



**HAL**  
open science

# Integer points close to a transcendental curve and correctly-rounded evaluation of a function

Nicolas Brisebarre, Guillaume Hanrot

► **To cite this version:**

Nicolas Brisebarre, Guillaume Hanrot. Integer points close to a transcendental curve and correctly-rounded evaluation of a function. 2021. hal-03240179v1

**HAL Id: hal-03240179**

**<https://hal.science/hal-03240179v1>**

Preprint submitted on 28 May 2021 (v1), last revised 21 Feb 2023 (v4)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# INTEGER POINTS CLOSE TO A TRANSCENDENTAL CURVE AND CORRECTLY-ROUNDED EVALUATION OF A FUNCTION

NICOLAS BRISEBARRE AND GUILLAUME HANROT

ABSTRACT. Despite several significant advances for the last 30 years, guaranteeing the correctly rounded evaluation of elementary functions, such as  $\cos$ ,  $\exp$ ,  $\sqrt[3]{\phantom{x}}$  for instance, is still a difficult issue. This is actually a Diophantine approximation problem, called the Table Maker's Dilemma, which consists in determining points with integer coordinates that are close to a curve. In this article, we propose two algorithmic approaches to tackle this problem. They are closely related to a celebrated work by Bombieri and Pila and to the so-called Coppersmith's method. We establish the underlying theoretical foundations, prove the algorithms, study their complexity and present practical experiments; we also compare our approach with previously existing approaches ones. In particular, our results show that the development of a correctly rounded mathematical library for the binary128 format is now possible at a much smaller cost than with previously existing approaches.

## CONTENTS

1. Introduction	2
1.1. Arithmetic framework	3
1.2. Correct rounding, Table Maker's Dilemma	4
1.3. Fast and cheap correctly-rounded function evaluation in binary64	5
1.4. Goal and outline of the paper	6
2. Formalization of the problem	7
3. State of the art	8
3.1. Diophantine approximation results: the exponential and the logarithm functions	9
3.2. Algorithmic approaches	10
4. A quick overview of uniform approximation and lattice basis reduction	10
4.1. Relation between uniform approximation and interpolation	10
4.2. A quick reminder on Euclidean lattices and the LLL algorithm	12
5. The one-variable method, à la Liouville	13
5.1. Volume estimates for rigorous interpolants at the Chebyshev nodes	13
5.2. Statement and proof of the algorithms	16
5.3. Proof of the algorithms	21
5.4. Practical remarks	21
5.5. Analysis	23
5.6. Complexity analysis	24

---

2020 *Mathematics Subject Classification*. Primary 11Y99, 41A05, 65G50; Secondary 11D75, 11J25.

This work was partly supported by the TaMaDi, FastRelax and NuSCAP projects of the French *Agence Nationale de la Recherche*.

6. The two-variable method	29
6.1. Volume estimates for rigorous interpolants at the Chebyshev nodes	30
6.2. Statement of the algorithms and proof of correctness	34
6.3. Analysis	36
6.4. Complexity analysis	37
7. Comparison with previous work	39
7.1. Univariate method and Bombieri and Pila’s approach	40
7.2. Bivariate method vs. Stehlé’s approach	40
8. Experimental results	41
8.1. Algorithms 1 and 2 in action: the TMD for the gamma function in binary128	41
8.2. Using $\omega_0$ in Algorithms 1 and 2 : the TMD for the exponential function in binary128 - experimental validation of Figure 1	42
8.3. Algorithms 3 and 4 in action: the TMD for the exponential function in binary128	42
9. Conclusion	44
Acknowledgements.	44
References	45
Appendix A. Proofs of facts regarding Chebyshev polynomials	48
Appendix B. Proof of Theorem 5.24	52
Appendix C. Lemmata on $\varphi, \psi$	53
Appendix D. Proofs of Theorems 6.4 and 6.13	56

## 1. INTRODUCTION

Modelling real numbers on a computer is by no means a trivial task. Until the mid-80s, processor manufacturers developed their own representations and conventions, leading to a difficult era – a time of weird, unexpected and dangerous behaviours [32]. This motivated the publication in 1984 of the IEEE-754 [17, 3] standard, since then revised in 2008 [28, 47] and 2019 [27], for binary floating-point (FP) arithmetic<sup>1</sup>, which remains the best trade-off for representing real numbers on a computer [50]. This put an end to this dangerous era of “numerical insecurity”.

In particular, the IEEE-754 standard clearly specifies the formats of the FP representations of numbers, and the behaviour of the four arithmetic operations and the square root. And yet, as of today, the standard still does not rule the behaviour of usual functions, such as the ones contained in the C mathematical library (`libm`), as precisely as it does for the four arithmetic operations and the square root.

The issue that we address in this paper is the problem of correctly-rounded evaluation of a one-variable function. Usually, when one wants to evaluate a function such as the cube root or the exponential functions, one actually evaluates a very good approximation of it (such as a polynomial for instance). This raises a problem, that is correct rounding: how can one guarantee that the rounding of the value of the function coincide with the rounding of the value of the approximation? This issue is related to a problem called Table’s Maker Dilemma (TMD).

---

<sup>1</sup>and the radix independent IEEE-854 [16, 4] standard that followed

This paper presents two heuristic approaches to address the TMD. Both mix results from approximation theory and algorithmic number theory (actually Euclidean lattice basis reduction). The first approach can be viewed as an effective variant of Bombieri and Pila's approach developed in [6] and the second one is an improvement over the algorithmic approaches developed in [61, 59]. In particular, it gives the first non-trivial results for the binary128 format and it paves the way for the first development of an efficient correctly rounded mathematical library in the three fundamental formats binary32, binary64 and binary128. As of today, the library `CRlibm`<sup>2</sup> [37] offers correctly rounded evaluation of the binary64 precision C99 standard elementary functions.

We believe that our results are interesting in themselves. In particular, beyond their application to the Table Maker's Dilemma for which we improve on some of the theoretical and practical results of [38, 40, 39, 61, 59], they offer a practical means to compute integer points in a strip around a transcendental analytic curve<sup>3</sup>.

**1.1. Arithmetic framework.** We first recall the definition of a FP number.

**Definition 1.1.** Let  $\beta, p, E_{\min}, E_{\max} \in \mathbb{Z}$ ,  $\beta, p \geq 2$ ,  $E_{\min} < 0 < E_{\max}$ , a (normal) radix- $\beta$  FP number in precision  $p$  with exponent range  $[E_{\min}, E_{\max}]$  is a number of the form

$$x = (-1)^s \frac{M}{\beta^{p-1}} \cdot \beta^E,$$

where :

- the *exponent*  $E \in \mathbb{Z}$  is such that  $E_{\min} \leq E \leq E_{\max}$ ,
- the *integral significand*  $M \in \mathbb{N}$  represented in radix  $\beta$  satisfies  $\beta^{p-1} \leq |M| \leq \beta^p - 1$ ,
- $s \in \{0, 1\}$  is the sign bit of  $x$ .

In the sequel, we shall leave the exponent range implicit unless it is explicitly required, and simply talk about “radix- $\beta$  FP numbers in precision  $p$ ”.

*Remark 1.2.* For the sake of clarity, we chose not to mention subnormal FP numbers since they will not appear in the text. One can find the complete definition in [50, Chap. 2.1].

The number zero is a special case, cf. [50, Chap. 3], that we add to the set of radix- $\beta$  and precision- $p$  FP numbers. This yields a set denoted  $\mathcal{F}_{\beta,p}$ .

*Remark 1.3.* In this paper, we use radix 2 for the sake of clarity but our approach remains valid for any radix, in particular radix 10, the importance of which grows at a steady pace.

Table 1 gives the main parameters of the three basic binary formats specified by IEEE 754-2008.

The result of an arithmetic operation whose input values belong to  $\mathcal{F}_p$  may not belong to  $\mathcal{F}_p$  (in general it does not). Hence that result must be rounded. The IEEE standard defines 5 different rounding functions; in the sequel,  $x$  is any real number to be rounded:

- round toward  $+\infty$ , or upwards:  $\circ_u(x)$  is the smallest element of  $\mathcal{F}_p$  that is greater than or equal to  $x$ ;

<sup>2</sup>[https://gforge.inria.fr/scm/browser.php?group\\_id=5929&extra=crlibm](https://gforge.inria.fr/scm/browser.php?group_id=5929&extra=crlibm)

<sup>3</sup>We mean here a representative curve of a transcendental analytic function.

	precision $p$	minimal exponent $E_{min}$	maximal exponent $E_{max}$
binary32	24	-126	127
binary64	53	-1022	1023
binary128	113	-16382	16383

TABLE 1. Main parameters of the three basic binary formats (up to 128 bits) specified by the standard [28].

- round toward  $-\infty$ , or downwards:  $\circ_d(x)$  is the largest element of  $\mathcal{F}_p$  that is less than or equal to  $x$ ;
- round toward 0:  $\circ_z(x)$  is equal to  $\circ_u(x)$  if  $x < 0$ , and to  $\circ_d(x)$  otherwise;
- round to nearest *ties to even*, denoted  $\circ_{ne}(x)$  and round to nearest *ties to away*, denoted  $\circ_{na}(x)$ . If  $x$  is exactly halfway between two consecutive elements of  $\mathcal{F}_p$ ,  $\circ_{ne}(x)$  is the one for which the integral significand  $j$  is an even number and  $\circ_{na}(x)$  is the one for which the integral significand  $j$  is largest. Otherwise, both return the element of  $\mathcal{F}_p$  that is the closest to  $x$ .

The first three rounding functions are called directed rounding functions.

The following numbers will play a key role in the problem that we address.

**Definition 1.4.** A rounding breakpoint (or simply, a breakpoint) is a value where the rounding function changes. For round-to-nearest functions, the rounding breakpoints are the exact middles of consecutive floating-point numbers. For the other rounding functions, they are the floating-point numbers themselves.

**1.2. Correct rounding, Table Maker’s Dilemma.** The standard requires that the user should be able to choose one rounding function among these ones, called the *active rounding function*. An active rounding function being chosen, when performing one of the 4 arithmetic operations, or when computing square roots, the obtained rounded result should be equal to the rounding of the exact result: this requirement on the quality of the computation is called *correct rounding*.

Being able to provide correctly rounded functions is of utter interest:

- it greatly improves the portability of software;
- it allows one to design algorithms that use this requirement;
- this requirement can be used for designing formal proofs of pieces of software;
- one can easily implement interval arithmetic, or more generally one can get certain lower or upper bounds on the exact result of a sequence of arithmetic operations.

While the IEEE 754-1985 and 854-1987 standards required correctly rounded arithmetic operations and square root, they did not do it for the most common mathematical functions, such as simple algebraic<sup>4</sup> functions like  $1/\sqrt{\cdot}$ ,  $\sqrt[3]{\cdot}$ ,  $\dots$  and also a few transcendental<sup>5</sup> functions like sine, cosine, exponentials, and logarithms of radices  $e$ , 2, and 10, etc. More generally, a natural target is the whole class of

<sup>4</sup>We say that a function  $\varphi$  is algebraic if there exists  $P \in \mathbb{Z}[x, y] \setminus \{0\}$  such that  $P(x, \varphi(x)) = 0$ .

<sup>5</sup>A function is transcendental if it is not algebraic.

elementary functions<sup>6</sup>. A subset of these functions is usually available from the `libms` delivered with compilers.

This lack of requirement is mainly due to a difficult problem known as the Table Maker's Dilemma (TMD), a term coined by Kahan. When evaluating most elementary functions, one has to compute some approximation to the exact result, using an intermediate precision somewhat larger than the "target" precision  $p$ . The TMD is the problem of determining, given a function  $f$ , what this intermediate precision should be in order to make sure that rounding that approximation yields the same result as rounding the exact result. Ideally, we aim at getting the minimal such precision<sup>7</sup>  $\text{htr}_f(p)$ , that we call *hardness to round* of  $f$  (see Definition 2.3).

Actually, if we have  $N$  FP numbers in the domain being considered, it is expected that  $\text{htr}_f(p)$  is of the order of  $p + \log_2(N)$  (hence  $2p$  for most usual functions and binades<sup>8</sup>). This is supported by a probabilistic heuristic approach that is presented in detail in [50, 49]. It has been studied in [10] where O. Robert and the authors of the present paper gave, under some mild hypothesis on  $f''$ , solid theoretical foundations to this probabilistic heuristic for certain ranges of values of  $k$ , targeting in particular the values of  $k$  that the `CRlibm` library uses in practice.

### 1.3. Fast and cheap correctly-rounded function evaluation in binary64.

Diophantine approximation-type methods yield – not fully satisfactory — upper bounds for  $\text{htr}_f(p)$  for algebraic functions: the precision to which the computations must be performed is, in general, grossly overestimated [29, 36, 11]. On the other hand, regarding transcendental functions, either no theoretical statement exists or they provide results that are off by such a margin that they cannot be used in practical computations [51, 34, 33].

Therefore, algorithmic approaches to the TMD [38, 40, 39, 61] had to be developed. They allowed for solving the TMD for the IEEE binary64 format (also known as "double precision").

As a consequence, the revised IEEE-754 standard now recommends (yet does not require, due to the lack of results in the case of binary128) that the following functions should be correctly rounded:  $e^x$ ,  $e^x - 1$ ,  $2^x$ ,  $2^x - 1$ ,  $10^x$ ,  $10^x - 1$ ,  $\ln(x)$ ,  $\log_2(x)$ ,  $\log_{10}(x)$ ,  $\ln(1+x)$ ,  $\log_2(1+x)$ ,  $\log_{10}(1+x)$ ,  $\sqrt{x^2+y^2}$ ,  $1/\sqrt{x}$ ,  $(1+x)^n$ ,  $x^n$ ,  $x^{1/n}$  ( $n$  is an integer),  $\sin(\pi x)$ ,  $\cos(\pi x)$ ,  $\arctan(x)/\pi$ ,  $\arctan(y/x)/\pi$ ,  $\sin(x)$ ,  $\cos(x)$ ,  $\tan(x)$ ,  $\arcsin(x)$ ,  $\arccos(x)$ ,  $\arctan(x)$ ,  $\arctan(y/x)$ ,  $\sinh(x)$ ,  $\cosh(x)$ ,  $\tanh(x)$ ,  $\sinh^{-1}(x)$ ,  $\cosh^{-1}(x)$ ,  $\tanh^{-1}(x)$ .

Thanks to these results, it is now possible to obtain correct rounding in binary64 in two steps only (inspired by a strategy developed by A. Ziv [70] and implemented in the `libultim` library<sup>9</sup>), which one may then optimize separately. This is the approach used in `CRlibm`:

- the first quick step is as fast as a current `libm`, and provides an accuracy of  $2^{-52-k}$  ( $k = 11$  for the exponential function for instance), which is sufficient to round correctly to the 53 bits of binary64 in most cases;

<sup>6</sup>An elementary function is a function of one variable which is the composition of a finite number of arithmetic operations ( $+$ ,  $-$ ,  $\times$ ,  $/$ ), exponentials, logarithms, constants, and solutions of algebraic equations [12, Def. 5.1.4].

<sup>7</sup>This notion will be formally defined in Section 2.

<sup>8</sup>A binade is an interval of the form  $[2^k, 2^{k+1})$  or  $(-2^{k+1}, -2^k]$  for  $k \in \mathbb{Z}$ .

<sup>9</sup>`libultim` was released by IBM. An updated version is now part of the GNU `glibc` and available under the GNU General Public License.

- the second accurate step is dedicated to challenging cases. It is slower but has a reasonable bounded execution time, being tightly targeted at the hardest-to-round cases computed by Lefèvre et al. [41, 40, 60, 61, 59]. In particular, there is no need for arbitrary multiple precision anymore.

This approach [21, 22] leads to correctly-rounded function evaluation routines that are fast and have a reasonable memory consumption. Unfortunately, the lack of useful information about the TMD in binary128 has so far prevented the development of an extension of `CRLibm` to this format.

**1.4. Goal and outline of the paper.** In this paper, we present two new algorithmic approaches to tackle the TMD. For both, we follow the standard strategy to subdivide the interval under study into subintervals; but instead of approximating the function  $f$  by a polynomial function using Taylor expansion at the center of such a tiny interval  $I$ , as it was done in [38, 40, 39, 61, 59], we approximate  $f$  by an algebraic function using uniform approximation: we search for  $P_1$  and  $P_2 \in \mathbb{Z}[X, Y]$  that are small on the curve  $(x, f(x))$  (first approach) or in a strip around this curve (second approach). This smallness implies that the bad cases for rounding are common roots to  $P_1$  and  $P_2$ . Then, we use a heuristic argument of Independence of  $P_1$  and  $P_2$ , analogous to the one used in [7, 61, 59] to obtain these bad cases.

In order to compute  $P_1$  and  $P_2$ , we use ideas and techniques developed by the first author and S. Chevillard [9]. Very roughly speaking, the key idea is to find  $P \in \mathbb{Z}[X_1, X_2]$  that is small at some points  $(x_i, f(x_i))$  of the curve (first approach) or  $(x_i, f(x_i) + y_i)$  of a strip around the curve (second approach). If the points  $x_i$  (resp. the pairs  $(x_i, y_i)$ ) are carefully chosen, these discrete smallness constraints imply uniform smallness over the curve, resp. the strip around the curve, cf. Section 4.1. The discrete constraints can be reformulated as the fact that the values of  $P$  at the  $(x_i, f(x_i))$ 's (resp. the  $(x_i, f(x_i) + y_i)$ 's) are the coordinates of a certain short vector in a Euclidean lattice. The celebrated LLL algorithm, cf. Section 4.2, then allows for computing a reasonable candidate for  $P$ .

The first approach which favours smallness on the curve  $(x, f(x)), x \in I$  is somehow akin to [6] whereas the second one, which forces smallness on a strip around this curve, is somehow analogous to [61, 59].

As our reader will see, our work does not lead, with respect to the previous algorithmic approaches, to an improvement for the determination of worst cases for rounding and optimal values of  $\text{htr}_f(p)$ . On the other hand, for certain elementary or special functions evaluated in binary128, we are able to provide:

- upper bounds for  $\text{htr}_f(p)$  that are useful in practice. For instance, for the exponential function, which plays a central role for correctly-rounded evaluation of the elementary functions of the C mathematical library, we provide a roadmap to reach, in practice, the upper bound  $\text{htr}_f(p) \leq 12p$  for  $p = 113$  which corresponds to the binary128 format;
- an effective determination of the FP values whose evaluation by  $f$  is exactly an FP number or the middle of two consecutive FP numbers. This is a key issue in the development of correctly-rounded evaluation routines. An exhaustive evaluation is possible in binary32, theoretical results [30] yield lists of these values for some restricted classes of functions and algorithmic approaches [41, 40, 60, 61, 59] make it possible to address this problem in binary64 but before this work, there was no practical means to tackle the

binary128 format. For instance, we address the case of the Euler function  $\Gamma$ , a function of the C mathematical library, for which theoretical results from Transcendental Number Theory are almost nonexistent.

This hopefully paves the way to an extension of `CRlibm` to the binary128 format, provided that we adopt the following three step strategy for this format:

- a first quick step identical to the one mentioned in the previous subsection;
- a second step, slower but with a reasonable bounded execution time, where the evaluation is performed using a precision of  $260 = 2p + 34$  bits, say. Heuristically, this should cover all the hardest-to-round cases;
- a third step, where the evaluation is performed using a precision of  $12p = 1356$  bits. Heuristically, this step should never be called, so it is important to write routines simple enough to be formally proved in order to guarantee its validity.

We will formalize the problem we address in Section 2. We then give a state of the art in Section 3. The theoretical results are presented in Section 3.1, including applications of [34, 33] that, to the best of our knowledge, are reviewed for the first time and offer theoretical upper bounds for  $\text{htr}_f(p)$  in the binary64 and binary128 cases which greatly improve upon the existing ones. The existing algorithmic approaches are sketched in Section 3.2. Our approach relies on tools from Approximation Theory and Euclidean lattice basis reduction and an idea presented in [9, 15]. We remind them in Section 4. Our first approach is presented in Section 5 and our second one in Section 6. We present a comparison with previous work in Section 7 and we conclude with experimental results in Section 8.

## 2. FORMALIZATION OF THE PROBLEM

Assume we wish to correctly round a real-valued function  $\varphi$ . We consider that all input values are elements of  $\mathcal{F}_p \cap [2^{e_1}, 2^{e_1+1})$ . The method must be applied for each possible value of  $e_1$ .

Note that if  $x$  is a bad case for  $\varphi$  (i.e.,  $\varphi(x)$  is difficult to round), then it is also a bad case for  $-\varphi$  and  $-x$  is a bad case for  $t \mapsto \varphi(-t)$  and  $t \mapsto -\varphi(-t)$ . Hence we can assume that  $x \geq 0$  and  $\varphi(x) \geq 0$ .

If the values of  $\varphi(x)$ , for  $x \in [2^{e_1}, 2^{e_1+1})$ , are not all included in the binade  $[2^{e_2}, 2^{e_2+1})$ , we split the input interval into subintervals such that for each subinterval, there is a rational integer  $e_2$  such that the values  $\varphi(x)$ , for  $x$  in the subinterval, are in  $[2^{e_2}, 2^{e_2+1})$ . We now restrict to one of those subintervals  $I$  included in  $[2^{e_1}, 2^{e_1+1})$ .

**For directed rounding functions**, the problem to be solved is the following:

**Problem 2.1** (TMD, directed rounding functions). *What is the minimum  $\mu(p) \in \mathbb{Z}$  such that, for  $2^{p-1} \leq X \leq 2^p - 1$  (and, possibly, the restrictions implied by  $X/2^{-e_1+p-1} \in I$ ) such that  $\varphi(X2^{e_1-p+1}) \notin \mathcal{F}_p$  and for  $2^{p-1} \leq Y \leq 2^p - 1$ , we have*

$$\left| 2^{-e_2} \varphi \left( \frac{X}{2^{-e_1+p-1}} \right) - \frac{Y}{2^{p-1}} \right| \geq \frac{1}{2^{\mu(p)}}.$$

**For rounding to nearest functions**, the problem to be solved is the following:

**Problem 2.2** (TMD, rounding to nearest functions). *What is the minimum  $\mu(p) \in \mathbb{Z}$  such that, for  $2^{p-1} \leq X \leq 2^p - 1$  (and, possibly, the restrictions implied by  $X/2^{-e_1+p-1} \in I$ ) such that  $\varphi(X2^{e_1-p+1})$  is not the middle of two consecutive*



elements of  $\mathcal{F}_p$  and for  $2^{p-1} \leq Y \leq 2^p - 1$ , we have

$$\left| 2^{-e_2} \varphi \left( \frac{X}{2^{-e_1+p-1}} \right) - \frac{2Y+1}{2^p} \right| \geq \frac{1}{2^{\mu(p)}}.$$

These statements lead to the following definition.

**Definition 2.3** (hardness to round). Let a precision  $p$  be given,  $\circ$  be a rounding function and  $\varphi$  be a real valued function. Let  $x$  be a FP number in precision  $p$  and  $e_2 \in \mathbb{Z}$  be the unique integer such that  $\varphi(x) \in [2^{e_2}, 2^{e_2+1})$  (here again, we assume  $x$  and  $\varphi(x) \geq 0$ , since the extension to the other cases is straightforward).

The hardness to round  $\varphi(x)$ , denoted  $\text{htr}_{\varphi, \{x\}, \circ}(p)$  is equal to:

- $-\infty$  if  $\varphi(x)$  is a breakpoint;
- the smallest integer  $m$  such that the distance of  $\varphi(x)$  to the nearest breakpoint is larger than or equal to  $2^{-m-e_2}$ .

The hardness to round  $\varphi$  over an interval  $I$ , denoted  $\text{htr}_{\varphi, I, \circ}(p)$ , is then the maximum of the hardness to round  $\varphi(x)$  for all FP  $x \in \mathcal{D}_p \cap I$ , while the hardness to round  $\varphi$  is the hardness to round  $\varphi$  over  $\mathbb{R}$ , simply denoted  $\text{htr}_{\varphi, \circ}(p)$ . When there is no ambiguity over the rounding function, we get rid of the symbol  $\circ$ .

*Remark 2.4.* Note that Problem 2.2 for precision  $p$  is a subproblem of Problem 2.1 for precision  $p+1$ .

*Remark 2.5.* If we assume that  $\varphi$  admits an inverse  $\varphi^{-1}$  and is differentiable over  $I$  and that we have a precise control over the image of  $\varphi'$  over  $I$ , it follows from the mean value theorem that addressing Problems 2.1 and 2.2 for  $\varphi$  over  $I$  is analogous to addressing Problems 2.1 and 2.2 for  $\varphi^{-1}$  over  $\varphi(I)$ . For instance, one can think of  $\exp$  and  $\log$  or  $x \mapsto \sqrt[3]{x}$  and  $x \mapsto x^3$ .

The problem that we actually tackle in this paper is the following.

**Problem 2.6.** Let  $a, b \in \mathbb{R}$ ,  $a < b$ ,  $f : [a, b] \rightarrow \mathbb{R}$  be a transcendental function analytic in a (complex) neighbourhood of  $[a, b]$ . Let  $u, v, w \in \mathbb{N} \setminus \{0\}$ , determine the integers  $X$ ,  $a \leq X/u \leq b$  for which there exists  $Y \in \mathbb{Z}$  satisfying

$$\left| f \left( \frac{X}{u} \right) - \frac{Y}{v} \right| < \frac{1}{w}.$$

This problem encompasses the TMD: consider  $a = 2^{e_1}$ ,  $b = 2^{e_1+1}$ ,  $u = 2^{p-e_1-1}$ ,  $v = 2^{p-e_2-1}$  and  $f = \varphi$  (for directed rounding functions) or  $f = \varphi + 1/(2v)$  (for rounding to nearest functions).

It also includes a generalization of the question addressed in [6], which corresponds to the case  $a = 0$ ,  $b = 1$ ,  $u = v$ .

### 3. STATE OF THE ART

In this section, we review the state of the art on the TMD; we shall start by discussing results which can be derived from previous estimates in Diophantine approximation, then shall account on the algorithmic approaches which have been developed since the late 90s. One can find a more complete (but slightly outdated since the results of [34, 33] on the exponential function are not considered) state of the art in [50, Chap. 12].

**3.1. Diophantine approximation results: the exponential and the logarithm functions.** We now follow the formalization that we presented in Section 2.

The exponential function is central in the study of correctly-rounded evaluation of the elementary functions of `libms`: a relevant information on its hardness to round yields relevant information as well on trigonometric and hyperbolic functions, and their respective reciprocals, see [50, §12.4.4] and Remark 2.5, the logarithm function and inverse trigonometric functions.

Following the works [51] and [34], Khémira and Voutier proved in [33] a lower bound (called transcendence measure) for the expression  $|e^\beta - \alpha|$ , where  $\alpha$  and  $\beta$  are algebraic numbers,  $\beta \neq 0$ . When specialized in FP numbers, their result provides interesting upper bounds for  $\text{htr}_{\text{exp}}(p)$ .

Let  $m$  and  $n \in \mathbb{N}$ , we put

$$d_n = \text{l.c.m.}(1, \dots, n) \text{ and } D_{m,n} = \frac{m!}{\prod_{\substack{q \leq n, \\ q \text{ prime}}} q^{v_q(m!)}}$$

where  $v_q(m!)$  is the  $q$ -adic valuation of  $m!$ . We can now state Khémira and Voutier's Theorem in the particular case where  $\alpha$  is a FP number and  $\beta$  is a FP number (directed rounded functions) or the middle of two consecutive FP numbers (round-to-nearest functions). As we mentioned above, we can get a similar result for the logarithm function.

**Theorem 3.1** (Khemira and Voutier [33], specialized here to FP numbers, directed rounded functions). *Let a precision  $p$  be given, let  $x \neq 0$  and  $y \in \mathcal{D}_p$  such that  $y$  and  $e^x$  are in the same binade. We denote by  $e_x$ , resp.  $e_y$ , the exponent of  $x$ , resp.  $y$ . We have  $e_y = \lfloor \log_2(\exp(x)) \rfloor = \lfloor x/\log(2) \rfloor$ . Let  $K$  and  $L \in \mathbb{N} \setminus \{0\}$ ,  $L \geq 2$  and  $E \in (1, +\infty)$  which satisfy to*

$$(3.1) \quad \begin{aligned} KL \log E &\geq KL \log 2 + (K-1)(1 + \log(\sqrt{3L}d_{L-1})) + \log(D_{K-1,L-1}) \\ &+ (1 + 2 \log 2)(L-1) + \log(\min(d_{L-2}^{K-1}, (L-2)!)) + \log((K-1)!) \\ &+ (K-1) \max(p-2-e_x, -1) \log 2 + LE|x| + L \log E \\ &+ (L-1) \max(p-2-e_y, -1) \log 2. \end{aligned}$$

Then we have  $|e^x - y| \geq E^{-KL}$ .

*Remark 3.2.* For the round-to-nearest functions, we assume that  $y$  is the middle of two consecutive FP numbers and that the numbers  $y$  and  $e^x$  are in the same binade. If we denote again  $e_y = \lfloor x/\log(2) \rfloor$ , the conclusion of the theorem remains valid if we replace  $\max(p-2-e_y, -1)$  with  $\max(p-1-e_y, -1)$  in the last line of (3.1).

*Remark 3.3.* The condition  $K \geq 2$  is not stated in [33] but it is actually necessary to have Inequality (3.1) satisfied here.

For instance, using the following sets of parameters, we are able to compute the following upper bounds for the hardness to round  $\exp$  on  $[1/2, 1)$ :

- In binary64 ( $p = 53$ ), the triple  $(K, L, E) = (78, 37, 52.4833)$  yields  $\text{htr}_{\text{exp}, [1/2, 1)}(53) \leq 16490 \sim 311p$ .
- In double extended precision ( $p = 64$ ), the triple  $(K, L, E) = (86, 41, 61.663)$  yields  $\text{htr}_{\text{exp}, [1/2, 1)}(64) \leq 20967 \sim 327p$ .
- In binary128 ( $p = 113$ ), the triple  $(K, L, E) = (122, 59, 83.32)$  yields  $\text{htr}_{\text{exp}, [1/2, 1)}(113) \leq 45928 \sim 406p$ .

**3.2. Algorithmic approaches.** In view of the lack of practicality (or in order to improve on it) of fundamental results discussed in the previous subsection, algorithmic approaches have been developed and used in an extensive way since the late 90s.

A first straightforward idea consists in testing all possible FP values  $x$ ; for each value of  $x$  one computes a sufficiently accurate interval approximation to  $f(x)$  and determines the hardness to round  $f(x)$ . The cost of the approach is obviously proportional to the number of different FP numbers of the format under study i.e.,  $2^{p+E_{\max}-E_{\min}}$ , which makes it basically tractable for binary32 [58]. In [23], the exhaustive evaluations are performed on an FPGA using a tabulated difference approach, which makes it possible to address the binary64 case. Currently, the double extended or binary128 formats seem completely out of reach of such approaches.

More subtle ideas proceed by splitting the domain into subintervals and replacing the function (assumed to be sufficiently smooth) by a polynomial, in practice a Taylor approximation, over the interval under study; one is then reduced to study the problem in the polynomial case.

Lefèvre, together with Muller [38, 40, 39], studied the degree 1 case; in this case, the remaining Diophantine problem is to find two integers  $x, y$ ,  $|x| \leq X$ ,  $|y| \leq Y$  such that  $|\alpha x + \beta - y|$  is minimal, which is solved by elementary Diophantine arguments, either the three distance theorem, or continued fractions (see e.g., [5]). These ideas lead to an algorithm of complexity  $\tilde{O}(2^{2p/3})$  for floating-point numbers of precision  $p$ , as  $p \rightarrow \infty$ , which computes all worst cases for rounding in the domains under consideration.

Highly optimized and parallel implementations of this method have proved invaluable to find optimal values of  $\text{htr}_f(53)$  for several functions of the standard `libm`. This was a key step towards the development of `CRlibm`.

Higher degree approximations give rise to more complicated Diophantine problems. Stehlé, Lefèvre and Zimmermann [61], further refined by Stehlé [59], make use of a technique due to Coppersmith [19, 20] and based on lattice basis reduction to solve it. We recall Corollaries 4 & 5 of [59], adapted to our context.

**Theorem 3.4** (Stehlé [59]). *For all  $\varepsilon > 0$ , there exists a heuristic algorithm of complexity  $\mathcal{P}(p)2^{p(1+\varepsilon)/2}$ , where  $\mathcal{P}$  is a polynomial, which, given a function  $f$ , returns all FP numbers  $x \in [1/2, 1)$  of precision  $p$  such that the hardness to round  $f(x)$  is  $\geq 2p$ .*

*There exists a polynomial-time heuristic algorithm which returns all FP numbers  $x \in [1/2, 1)$  of precision  $p$  such that the hardness to round  $f(x)$  is  $\geq 4p^2$ ; the latter works by reducing a lattice of dimension  $O(p^2)$  of  $\mathbb{R}^m$  for some  $m = O(p^4)$ .*

The heuristic character of the algorithm is rather mild (i.e., the algorithm works in practice as expected on almost all inputs).

#### 4. A QUICK OVERVIEW OF UNIFORM APPROXIMATION AND LATTICE BASIS REDUCTION

We shall require some tools from uniform approximation theory [24, 57, 56, 14, 8, 48, 65] and algorithmic geometry of numbers [46, 26, 18, 13, 53, 67].

**4.1. Relation between uniform approximation and interpolation.** Let  $n \in \mathbb{N}$ , the  $n$ -th Chebyshev polynomial of the first kind is defined by  $T_n \in \mathbb{R}_n[x]$  and

$T_n(\cos t) = \cos(nt)$  for all  $t \in [0, \pi]$ . The  $T_n$ 's can also be defined by

$$T_0(x) = 1, T_1(x) = x, T_{n+2}(x) = 2xT_{n+1}(x) - T_n(x), \forall n \in \mathbb{N}.$$

4.1.1. *Interpolation at the Chebyshev nodes.* The zeros of  $T_{n+1}$  are

$$\mu_{k,n} = \cos\left(\frac{(n-k+1/2)\pi}{n+1}\right), k = 0, \dots, n.$$

They are called  $(n+1)$ -Chebyshev nodes of the first kind. Polynomials interpolating functions at this family give rise to very good uniform approximations to these functions [48, 65]. To be able to work on an interval  $[a, b]$ , we will need scaled versions of Chebyshev polynomials and nodes. We then define, for  $n \in \mathbb{N}$ ,

$$(4.1) \quad T_{n,[a,b]} := T_n\left(\frac{2x-b-a}{b-a}\right), \mu_{k,n,[a,b]} := \frac{(b-a)\mu_k + a + b}{2} \text{ for } k = 0, \dots, n.$$

Here again, when there is no ambiguity, we denote the nodes as  $\mu_{k,[a,b]}$ . Note that  $T_{n,[a,b]}(\mu_{k,[a,b]}) = T_n(\mu_k)$ .

Let  $N \geq 1$ , let  $f$  be a function defined over  $[a, b]$ , if we interpolate  $f$  by a polynomial in  $\mathbb{R}_{N-1}[x]$  at the scaled Chebyshev nodes of the first kind, we have the following expressions for the interpolation polynomial  $P$  [48, Chap. 6]:

$$\begin{aligned} P(x) &= \sum'_{0 \leq k \leq N-1} c_k T_{k,[a,b]}(x) \in \mathbb{R}_{N-1}[x] \text{ with} \\ c_k &= \frac{2}{N} \sum_{0 \leq \ell \leq N-1} f(\mu_{\ell,[a,b]}) T_{k,[a,b]}(\mu_{\ell,[a,b]}) \text{ for } k = 0, \dots, N-1, \\ &= \frac{2}{N} \sum_{0 \leq \ell \leq N-1} f(\mu_{\ell,[a,b]}) T_k(\mu_{\ell}). \end{aligned}$$

The symbol  $\sum'$  means that the first coefficient has to be halved. Note that, if we introduce  $\hat{f} : z \in [-1, 1] \mapsto f\left(z\frac{b-a}{2} + \frac{a+b}{2}\right)$ , the coefficients  $c_k$  are also the coefficients of the interpolation polynomial in  $\mathbb{R}_{N-1}[x]$  of  $\hat{f}$  at the Chebyshev nodes of the first kind.

4.1.2. *Uniform approximation using interpolation polynomials.* Let  $\rho > 1$ , we define the ellipsis

$$(4.2) \quad \mathcal{E}_\rho = \left\{ \frac{\rho e^{i\theta} + \rho^{-1} e^{-i\theta}}{2}, \theta \in [0, 2\pi] \right\}$$

and  $E_\rho$  be the closed region bounded by  $\mathcal{E}_\rho$ . Let  $N \in \mathbb{N}$ ,  $N \geq 1$ , we also define

$$\eta_{\rho,0,N-1} = 1 \text{ and } \eta_{\rho,k,N-1} = \frac{\rho^2 + 1}{\rho^2 - 1} \text{ for } k = 1, \dots, N-1.$$

The following two propositions establish Cauchy's inequalities for interpolation polynomials at Chebyshev nodes.

**Proposition 4.1.** *Let  $\rho > 1$ , let  $N \in \mathbb{N}$ ,  $N \geq 1$ ,  $f$  be a function analytic in a neighbourhood of  $E_\rho$ , the coefficients  $c_k, k = 0, \dots, N-1$ , of the interpolation polynomial  $p_{N-1}$  of  $f$  at the Chebyshev nodes of the first kind  $(\mu_{k,N-1})_{0 \leq k \leq N-1}$  satisfy*

$$|c_k| \leq 2 \frac{M_\rho(f)}{\rho^k} \eta_{\rho,k,N-1} \text{ for } k = 0, \dots, N-1,$$

where  $M_\rho(f) = \max_{z \in \mathcal{E}_\rho} |f(z)|$ . Moreover, we have

$$\|f - p_{N-1}\|_{\infty, [-1,1]} \leq \frac{4M_\rho(f)}{\rho^{N-1}(\rho - 1)}.$$

*Proof.* See Appendix A.  $\square$

Let  $\rho_1, \rho_2 > 1$ , we define  $\mathcal{E}_{\rho_1, \rho_2} = \mathcal{E}_{\rho_1} \times \mathcal{E}_{\rho_2}$  and  $E_{\rho_1, \rho_2} = E_{\rho_1} \times E_{\rho_2}$ .

**Proposition 4.2.** *Let  $\rho_1, \rho_2 > 1$ , let  $M_1, M_2 \in \mathbb{N}, M_1, M_2 \geq 2$ ,  $f$  be a function analytic in a neighbourhood of  $E_{\rho_1, \rho_2}$ , the coefficients  $c_{k_1, k_2}, k_1 = 0, \dots, M_1 - 1, k_2 = 0, \dots, M_2 - 1$  of the interpolation polynomial  $P_{M_1-1, M_2-1}$  of  $f$  at pairs of Chebyshev nodes of the first kind satisfy, for  $k_1 = 0, \dots, M_1 - 1, k_2 = 0, \dots, M_2 - 1$ ,*

$$|c_{k_1, k_2}| \leq 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{k_1} \rho_2^{k_2}} \eta_{\rho_1, k_1, M_1-1} \eta_{\rho_2, k_2, M_2-1},$$

where  $M_{\rho_1, \rho_2}(f) = \max_{z \in \mathcal{E}_{\rho_1, \rho_2}} |f(z)|$ . Moreover, we have

$$\|f - P_{M_1-1, M_2-1}\|_{\infty, [-1,1] \times [-1,1]} \leq \frac{16\rho_1\rho_2 M_{\rho_1, \rho_2}(f)}{(\rho_1 - 1)(\rho_2 - 1)} \left( \frac{1}{\rho_1^{M_1}} + \frac{1}{\rho_2^{M_2}} \right).$$

*Proof.* See Appendix A.  $\square$

#### 4.2. A quick reminder on Euclidean lattices and the LLL algorithm.

In this subsection, we review shortly basic facts concerning lattices and lattice basis reduction algorithms.

**Definition 4.3.** Let  $M \in \mathbb{N}, M \geq 1$ , a lattice of  $\mathbb{R}^M$  is a discrete subgroup of  $\mathbb{R}^M$ ; equivalently, a lattice  $L \subset \mathbb{R}^M$  is the set of integer linear combinations of a family  $(b_1, \dots, b_N)$  of  $\mathbb{R}$ -linearly independent vectors of  $\mathbb{R}^M$ . We shall then say that  $(b_i)_{1 \leq i \leq N}$  is a basis of  $L$ , and that  $N \leq M$  is the dimension of  $L$ .

Given a basis  $(b_1, \dots, b_N)$  of  $L$  as input, finding a shortest nonzero vector in  $L$  is called the *shortest vector problem*. The decision version of this problem has been shown [1] to be hard under randomized reductions; in practice, one thus has to content oneself with approximation algorithms, most often the LLL algorithm [42]:

**Theorem 4.4** (Lenstra, Lenstra, Lovász, 1982). *The LLL algorithm, given  $N$   $\mathbb{R}$ -linearly independent vectors  $(b_1, \dots, b_N) \in \mathbb{Z}^M$ , returns in time polynomial in the maximal bit-length of the coefficients of the  $b_i$ 's, the lattice rank  $N$ , and the space dimension  $M$ , a basis  $(c_1, \dots, c_N)$  such that  $\|c_1\|_2 \leq 2^{(N-1)/4} (\text{vol}(L))^{1/N}$ , and  $\|c_1\|_2 \leq 2^{(N-1)/2} \min_{x \in L - \{0\}} \|x\|_2$ . One also has  $\|c_2\|_2 \leq 2^{(N-1)/4} (\text{vol}(L))^{1/(N-1)}$ .*

*Proof.* See Theorems 9 and 10 from [53, Chap. 2], except for the last inequality which is a consequence of the proof of Fact 3.3 in [7].  $\square$

The constant 2 of the inequalities of the theorem is arbitrary, and could be replaced by any real number  $> 4/3$ .

We now discuss shortly an improvement due to Akhavi & Stehlé [2] to the LLL algorithm in the case where  $N$  is much smaller than  $M$ , the dimension of the ambient space. Let  $A$  be a  $N \times M$  matrix, the rows of which generate the lattice  $L$ ; the idea is to reduce a smaller  $N \times N$  matrix obtained by a random projection (i.e., multiplying  $A$  on the right by a random  $M \times N$  matrix), and apply the same transformation to the original matrix.

**Theorem 4.5** (Akhavi & Stehlé, 2008). *For all  $N$ , there is an  $n_0(N)$  such that for  $n \geq n_0(N)$ , if  $P$  is a matrix whose entries are independent random vectors picked up uniformly independently inside the  $N$ -th dimensional unit ball, and  $A' = \text{LLL}(A \cdot P)$ ; then, with probability  $\geq 1 - 2^{-d}$ , the first column vector  $c_1$  of the matrix  $A(A \cdot P)^{-1}A'$  is a vector of  $L$  of norm  $\leq 2^{4N}(\det L)^{1/N}$ .*

S. Torres [64] showed that this idea indeed improves the practical outcome of [59]. As for us, cf. Section 8, we noticed that this idea works in practice much better than the theorem suggests, and simpler models for the random matrix  $P$  such as random, uniform  $\{0, \pm 1\}$  coefficients give equally good results.

## 5. THE ONE-VARIABLE METHOD, À LA LIOUVILLE

Let  $u, v \in \mathbb{N} \setminus \{0\}$ ,  $a < b$  be two real numbers and  $f : [a, b] \rightarrow \mathbb{R}$ . The starting point of this approach follows a simple (but fundamental!) idea due to J. Liouville [43, 44, 45], which we now recall.

For any  $P \in \mathbb{Z}[X_1, X_2]$ , for any  $x \in [a, b]$ ,  $y \in [f(x) - 1/v, f(x) + 1/v]$ , we know, from the mean value theorem, that there exists  $c$  between  $vf(x)$  and  $vy$  (hence  $c \in [vf(x) - 1, vf(x) + 1]$ ), such that

$$(5.1) \quad P(ux, vf(x)) - P(ux, vy) = v(f(x) - y) \frac{\partial P}{\partial y}(ux, c).$$

Then we compute, by combining Chebyshev interpolation and lattice reduction, two polynomials  $P_0, P_1 \in \mathbb{Z}[X_1, X_2]$  such that, for  $i = 0, 1$ ,  $|P_i(ux, vf(x))| < 1/2$ , while  $\left| \frac{\partial P_i}{\partial y}(ux, c) \right|$  is bounded by “not too large” a  $M$ . Let  $x_0 = X/u \in [a, b]$ ,  $X \in \mathbb{Z}$ ,  $y_0 = Y/v$ ,  $Y \in \mathbb{Z}$ , we have  $P_i(ux_0, vy_0) = P_i(X, Y) \in \mathbb{Z}$ . If, for instance  $P_0(X, Y)$  is non zero, then it follows from (5.1)

$$\underbrace{|P_0(X, Y)|}_{\geq 1} - \underbrace{|P_0(X, vf(X/u))|}_{< 1/2} \leq v|f(X/u) - Y/v| \underbrace{\left| \frac{\partial P_0}{\partial y}(ux_0, c) \right|}_{\leq M},$$

hence  $|vf(X/u) - Y| > 1/(2M)$ . Otherwise, we have  $P_0(X, Y) = P_1(X, Y) = 0$ . We now use our heuristic assumption, that is  $P_0$  and  $P_1$  have no nonconstant common factor: we then perform elimination of one of the variables and retrieve the list of all the bad cases, i.e., the  $X$  such  $|vf(X/u) - Y| \leq 1/(2M)$ . In the sequel of this section, we give all the details of this approach: we first give estimates of the determinants of the lattices that we use, we present our algorithm, the proof of its correctness and analyse its complexity.

### 5.1. Volume estimates for rigorous interpolants at the Chebyshev nodes.

Let  $N \geq 2$ , for  $i = 0, \dots, N-1$ , let  $f_i$  be a function defined over  $[a, b]$  and  $Q_i$  be its interpolation polynomials in  $\mathbb{R}_{N-1}[x]$  at Chebyshev nodes of the first kind.

Let DCT-II denote the discrete cosine transform of type 2:

$$\text{DCT-II} : \mathbb{R}^N \rightarrow \mathbb{R}^N$$

$$(x_0, \dots, x_{N-1}) \mapsto (X_0, \dots, X_{N-1}) \text{ with}$$

$$X_k = \sum_{0 \leq \ell \leq N-1} x_\ell \cos\left(\frac{k(\ell + 1/2)\pi}{N}\right), \text{ for } k = 0, \dots, N-1.$$

This function is often introduced with slightly different normalizations [62, 55] and we can take advantage of fast algorithms that make it possible to compute it in at most  $\mathcal{O}(N \log N)$  operations. Recall, cf. Section 4.1.1, that for  $i = 0, \dots, N-1$ ,

$$Q_i(x) = \sum'_{0 \leq k \leq N-1} c_{k,i} T_{k,[a,b]}(x) \in \mathbb{R}_{N-1}[x] \text{ with} \\ (c_{0,i}, \dots, c_{N-1,i}) = \frac{2}{N} \text{DCT-II}(f_i(\mu_{N-1,[a,b]}), \dots, f_i(\mu_{0,[a,b]})).$$

Let us introduce two real parameters  $\rho > 1$  and  $\omega_0 \geq 1$  (to be chosen later on). Let  $a < b$ , we denote  $\mathcal{E}_{\rho,a,b} = \left\{ \frac{b-a}{2} \frac{\rho e^{i\theta} + \rho^{-1} e^{-i\theta}}{2} + \frac{a+b}{2}, \theta \in [0, 2\pi] \right\}$  and  $E_{\rho,a,b}$  be the closed region bounded by the ellipse  $\mathcal{E}_{\rho,a,b}$ .

We now assume that all the  $f_i$ 's are analytic in a neighbourhood of  $E_{\rho,a,b}$ . For  $i = 0, \dots, N-1$ , let  $R_i = 4M_{\rho,a,b}(f_i)/(\rho^{N-1}(\rho-1))$ . We have, cf. Proposition 4.1,  $\|f_i - Q_i\|_{\infty,[a,b]} \leq R_i$ . If  $\delta_{ij}$  denotes the Kronecker delta, we introduce the  $N \times 2N$  matrix  $A = (A_1 A_2)$ , where

$$(5.2) \quad A_1 = (c_{j,i}/2^{\delta_{j0}})_{0 \leq i,j \leq N-1}, \quad A_2 = (\delta_{ij} \rho^{\omega_0} R_i)_{0 \leq i,j \leq N-1}.$$

Its rows generate the lattice that will be reduced in our algorithm. The (diagonal) right half of the matrix are weights that will be used for two tasks:

- (remainders) controlling that  $P_0(ux, vf(x))$ ,  $P_1(ux, vf(x))$  are uniformly small, where  $P_0, P_1$  are two polynomials output by the lattice reduction process; this accounts for the presence of the  $R_i$  term;
- (coefficients) controlling the size of the coefficients of the polynomial, hence the quality of the lower bound produced by Algorithm 2; this accounts for the presence of the  $\rho^{\omega_0}$  term. We shall assume later on (in Section 5.2) that  $\rho^{\omega_0} < \rho^{N-1}(\rho-1)/4$ . This is a necessary condition for the success of the method; otherwise, all the  $\rho^{\omega_0} R_i$  are too large, and the method is bound to fail.

We now establish a slightly improved version of [59, Theorem 2]<sup>10</sup>. Let an  $N \times M$  matrix  $B$  whose rows span a lattice  $L$ . We assume that the entries of  $B$  satisfy:  $|B_{i,j}| \leq r_i \cdot c_j$ , for some  $r_i$ 's and  $c_j$ 's,  $0 \leq i \leq N-1$  and  $0 \leq j \leq M-1$ . As mentioned in [59], this is typical for Coppersmith-type lattice bases.

**Theorem 5.1.** *Let  $B$  be an  $N \times M$  matrix (with  $M \geq N$ ), the entries of which are bounded by the product of some quantities  $r_i$ 's and  $c_j$ 's as described above. Let  $L$  be the lattice spanned by the rows of the matrix  $B$ , and  $\mathfrak{P}$  the product of the  $N$  largest  $c_j$ 's. We have:*

$$\text{vol } L = (\det B^t B)^{1/2} \leq \binom{M}{N}^{1/2} N^{N/2} \left( \prod_{i=0}^{N-1} r_i \right) \mathfrak{P}.$$

*Proof.* Let us denote  $\mathbf{C}_0, \dots, \mathbf{C}_{M-1}$  the columns of  $B$ . The classical Lagrange's identity, which is a particular case of Cauchy-Binet formula [25, 35], then states

<sup>10</sup>Note that there is an inaccuracy in the statement of [59, Theorem 2]:  $\sqrt{n}$  should be replaced with  $\binom{n}{d}^{1/2} d^{1/2}$ .

that

$$(5.3) \quad \det B {}^t B = \sum_{0 \leq j_1 < \dots < j_N \leq M-1} \det(\mathbf{C}_{j_1}, \dots, \mathbf{C}_{j_N})^2 \\ \leq \binom{M}{N} \max_{0 \leq j_1 < \dots < j_N \leq M-1} \det(\mathbf{C}_{j_1}, \dots, \mathbf{C}_{j_N})^2.$$

We can assume  $\left(\prod_{i=0}^{N-1} r_i\right) \neq 0$ : otherwise, there is at least one row of  $B$  that is identically 0, hence  $\text{vol } L = 0$ .

Now, for a given  $0 \leq j_1 < \dots < j_N \leq M-1$ , if one of the  $c_j$  is zero, it follows that at least one column of  $(\mathbf{C}_{j_1}, \dots, \mathbf{C}_{j_N})$  is zero, hence  $\det(\mathbf{C}_{j_1}, \dots, \mathbf{C}_{j_N}) = 0$ . Otherwise, we consider the matrix  $(\mathbf{C}'_{j_1} \dots \mathbf{C}'_{j_N})$  obtained from  $(\mathbf{C}_{j_1} \dots \mathbf{C}_{j_N})$  after having divided the  $i$ -th row by  $r_i$  for all  $i = 0, \dots, N-1$  and the  $j_k$ -th column by  $c_{j_k}$  for all  $k = 1, \dots, N$ . All the coefficients of  $(\mathbf{C}'_{j_1} \dots \mathbf{C}'_{j_N})$  have an absolute value lesser or equal to 1. Hadamard's inequality then implies  $\det(\mathbf{C}'_{j_1}, \dots, \mathbf{C}'_{j_N})^2 \leq N^N$ . It follows

$$\det(\mathbf{C}_{j_1}, \dots, \mathbf{C}_{j_N})^2 = \left(\prod_{i=0}^{N-1} r_i\right)^2 \left(\prod_{k=1}^N c_{j_k}\right)^2 \det(\mathbf{C}'_{j_1}, \dots, \mathbf{C}'_{j_N})^2 \\ \leq \left(\prod_{i=0}^{N-1} r_i\right)^2 \mathfrak{P}^2 N^N.$$

We conclude by combining the last inequality with (5.3).  $\square$

Then, we upper bound  $\det A {}^t A$ .

**Theorem 5.2.** *Let  $\rho > 1$ ,  $\omega_0 \geq 0$ ,  $a < b$ ,  $N \geq 2$ ,  $f_0, \dots, f_{N-1}$  be functions analytic in a neighbourhood of  $E_{\rho, a, b}$ . We have*

$$(\det A {}^t A)^{1/2} \leq (32N)^{N/2} \left(\frac{\rho}{\rho-1}\right)^N \frac{\prod_{i=0}^{N-1} M_{\rho, a, b}(f_i)}{\rho^{N(N-1)/2 + [\omega_0]([\omega_0] - 2\omega_0 + 1)/2}}$$

where  $M_{\rho, a, b}(f) = \max_{z \in E_{\rho, a, b}} |f(z)|$ .

*Proof.* We introduce, for all  $i = 0, \dots, N-1$ ,  $\widehat{f}_i : z \mapsto f_i(z^{\frac{b-a}{2}} + \frac{a+b}{2})$ , which, by assumption, is analytic in a neighbourhood of  $E_\rho$ .

For  $0 \leq i, j \leq N-1$ , we have from Proposition 4.1 and Lemma A.6,

$$|A_{1, i, j}| \leq \left| \frac{c_{j, i}}{2^{\delta_{j0}}} \right| \leq 2 \underbrace{M_\rho(\widehat{f}_i)}_{=M_{\rho, a, b}(f_i)} \frac{1}{\rho^j} \frac{\rho^2 + 1}{\rho^2 - 1} \leq 2 \frac{M_{\rho, a, b}(f_i)}{\rho^j} \frac{\rho}{\rho - 1}.$$

Now, let us write, for  $0 \leq i, j \leq N-1$ ,

$$|A_{2, i, j}| \leq \rho^{\omega_0} R_i = 4\rho^{\omega_0} \frac{M_{\rho, a, b}(f_i)}{\rho^{N-1}(\rho-1)} = 4\rho^{\omega_0} \frac{\rho}{\rho-1} \frac{M_{\rho, a, b}(f_i)}{\rho^N}.$$

Consequently, we can put  $r_i = 2\rho/(\rho-1)M_{\rho, a, b}(f_i)$  for  $i = 0, \dots, N-1$ . Now, in order to apply Theorem 5.1, we determine the product  $\mathfrak{P}$  of the  $N$  largest elements among

$$1, \frac{1}{\rho}, \dots, \frac{1}{\rho^{N-1}}, \underbrace{\frac{2\rho^{\omega_0}}{\rho^N}, \dots, \frac{2\rho^{\omega_0}}{\rho^N}}_{N \text{ times}}.$$



Hence,  $\mathfrak{F}$  is the maximum, for  $s = 1, \dots, N$ , of

$$\prod_{k=1}^s \frac{1}{\rho^{k-1}} \prod_{k=s+1}^N \frac{2\rho^{\omega_0}}{\rho^N} = \frac{1}{\rho^{s(s-1)/2}} \frac{(2\rho^{\omega_0})^{N-s}}{\rho^{N(N-s)}},$$

Finally, Theorem 5.1 yields

$$\begin{aligned} (\det A {}^t A)^{1/2} &\leq \binom{2N}{N}^{1/2} N^{N/2} 2^N \left( \frac{\rho}{\rho-1} \right)^N \max_{1 \leq s \leq N} \frac{(2\rho^{\omega_0})^{N-s}}{\rho^{s(s-1)/2+N(N-s)}} \prod_{i=0}^{N-1} M_{\rho,a,b}(f_i) \\ &\leq (32N)^{N/2} \left( \frac{\rho}{\rho-1} \right)^N \max_{1 \leq s \leq N} \frac{\rho^{\omega_0(N-s)}}{\rho^{s(s-1)/2+N(N-s)}} \prod_{i=0}^{N-1} M_{\rho,a,b}(f_i). \end{aligned}$$

Then, if  $P_s = \rho^{\omega_0(N-s)}/\rho^{s(s-1)/2+N(N-s)}$ , we have  $P_{s+1}/P_s = \rho^{N-s-\omega_0}$ , hence  $P_s$  is maximal for  $s = N - \lfloor \omega_0 \rfloor$ , which completes the proof of the Theorem.  $\square$

We now specialize the previous estimate to our situation, where we shall use the ordered list of functions  $[f_i, 0 \leq i \leq (d+1)(d+2)/2 - 1] = [x \mapsto u^k x^k v^\ell f(x)^\ell, \ell = 0, \dots, d, k = 0, \dots, d - \ell]$ .

**Corollary 5.3.** *Let  $\rho > 1$ ,  $a < b$ ,  $f$  be a function analytic in a neighbourhood of  $E_{\rho,a,b}$ . Let  $d \geq 1$ ,  $N = (d+1)(d+2)/2$ ,  $\omega_0 \geq 0$ ,  $u, v \in \mathbb{N} \setminus \{0\}$ . Define  $f_{k,\ell}(x) = u^k x^k v^\ell f(x)^\ell$  for  $0 \leq \ell \leq d$ ,  $0 \leq k \leq d - \ell$ , the matrices  $A_1, A_2, A = (A_1 A_2)$  as in (5.2), and the quantity  $\Delta_{N,[a,b],\omega_0} := (\det A {}^t A)^{1/2}$ . We have*

$$(5.4) \quad \Delta_{N,[a,b],\omega_0}^{1/(N-1)} \leq 18\sqrt{N} \left( \frac{\rho}{\rho-1} \right)^{N/(N-1)} \frac{(uv)^{2N/(3(d+3))}}{\rho^{N/2 + \lfloor \omega_0 \rfloor (\lfloor \omega_0 \rfloor - 2\omega_0 + 1)/(2(N-1))}} \left( \frac{b-a}{2} \left( \frac{\rho + \rho^{-1}}{2} \right) + \left| \frac{b+a}{2} \right| \right)^{2N/(3(d+3))} M_{\rho,a,b}(f)^{2N/(3(d+3))}.$$

*Proof.* Follows from Theorem 5.2, the facts that  $M_{\rho,a,b}(x^k f(x)^\ell) \leq M_{\rho,a,b}(x)^k M_{\rho,a,b}(f)^\ell$ , the inequality

$$M_{\rho,a,b}(x) \leq \frac{b-a}{2} \left( \frac{\rho + \rho^{-1}}{2} \right) + \left| \frac{b+a}{2} \right|,$$

and finally the fact that  $(4\sqrt{2N})^{N/(N-1)} \leq 18\sqrt{N}$  for  $N \geq 3$ .  $\square$

**5.2. Statement and proof of the algorithms.** Our main routine is Algorithm 2. It comes together with Algorithm 1 that mainly constructs the lattice to be reduced in Algorithm 2. To read the last algorithm, we need to introduce one notation: for any  $x \in \mathbb{R}$ ,  $[x]_0 = \max([x], -[-x])$ .

Let  $[R_{2,i}, i = 0, \dots, N-1]$  be the ordered list  $[4 \frac{u^k M_{\rho,a,b}(x)^k v^\ell M_{\rho,a,b}(f)^\ell}{\rho^{N-1}(\rho-1)}; \ell = 0, \dots, d, k = 0, \dots, d - \ell]$ . Let  $A = (A_1 A_2)$  and  $\hat{A} = (\hat{A}_1 \hat{A}_2)$  be the  $N \times 2N$  matrices defined by

$$\begin{aligned} A_1 &= (c_{j,i}/2^{\delta_{j^0}})_{0 \leq i,j \leq N-1}, & A_2 &= (\delta_{ij} \rho^{\omega_0} R_{2,i})_{0 \leq i,j \leq N-1}, \\ \hat{A}_1 &= ([2^{\text{prec}} A_1[i,j]]_0 / 2^{\text{prec}})_{0 \leq i,j \leq N-1}, & \hat{A}_2 &= ([2^{\text{prec}} A_2[i,j]] / 2^{\text{prec}})_{0 \leq i,j \leq N-1}, \end{aligned}$$

where  $\text{prec} = \lceil -\log_2(\min_{0 \leq i \leq N-1} A_2[i,i]) + \log_2(N) \rceil + 5$ . From a mathematical point of view, the lattice that we would ideally work with is the one generated by the rows of  $A$ . And yet, in order to perform lattice reduction computations, it is

**Algorithm 1** Computation of the lattice to be reduced (1D approach)

**Input:** Two real numbers  $a < b$ ,  $f$  a transcendental function analytic in a complex neighbourhood of  $[a, b]$ , three nonzero integers  $d, u, v$ , two real numbers  $\rho > 1, \omega_0$  such that  $1 \leq \rho^{\omega_0} < \rho^{N-1}(\rho - 1)/4$ , where  $N = (d + 1)(d + 2)/2$ .

**Output:** Two matrices  $M_c, M_r \in \mathcal{M}_N(\mathbb{Z})$ , respectively storing scaled values of the coefficients and the remainders, an integer **prec** which is the precision of the computations.

```

1:  $N \leftarrow (d + 1)(d + 2)/2, i \leftarrow 0$ 
   // Computation of the Chebyshev nodes, listed in reverse order
2:  $L_{cheb} \leftarrow \left[ \frac{b-a}{2} \cos\left(\left(j + \frac{1}{2}\right)\frac{\pi}{N}\right) + \frac{a+b}{2} \right]_{0 \leq j \leq N-1}$ 
3:  $M_c \leftarrow [0]_{N \times N}; M_r \leftarrow [0]_{N \times N}$ 
4:  $B_x \leftarrow \left\lfloor \frac{a+b}{2} \right\rfloor + \frac{b-a}{4}(\rho + \rho^{-1})$ 
5:  $g \leftarrow (x \mapsto \left| f\left(\frac{a+b}{2} + \frac{b-a}{4}(\rho \exp(ix) + \rho^{-1} \exp(-ix))\right) \right|)$ 
6:  $B_f \leftarrow \max(g([0, 2\pi]))$ 
7:  $\text{prec} \leftarrow \lceil (N - 1 - \omega_0) \log_2(\rho) + \log_2(\rho - 1) + \log_2(N) \rceil + 3$ 
8: for  $\ell = 0$  to  $d$  do
9:   for  $k = 0$  to  $d - \ell$  do
10:     $\varphi \leftarrow (x \mapsto (ux)^k (vf(x))^\ell)$ 
       // We compute the coefficient matrix : for each function, we compute its
       // value at points of  $L_{cheb}$ , use DCT and scale.
11:     $U \leftarrow [\varphi(L_{cheb}[0]), \dots, \varphi(L_{cheb}[N - 1])]$ 
12:     $L_{\text{DCT}} \leftarrow \frac{2}{N} \text{DCT-II}(U), L_{\text{DCT}}[0] \leftarrow \frac{1}{2} L_{\text{DCT}}[0]$ 
13:    for  $j = 0$  to  $N - 1$  do
14:       $M_c[i, j] \leftarrow \lceil 2^{\text{prec}} L_{\text{DCT}}[j] \rceil_0$  // Scaling of the coefficient matrix
15:    end for
       // We compute the scaled remainder matrix.
16:     $M_r[i, i] \leftarrow \left\lfloor 2^{\text{prec}} \frac{4\rho^{\omega_0}}{\rho^{N-1}(\rho-1)} (uB_x)^k (vB_f)^\ell \right\rfloor, i \leftarrow i + 1$ 
17:    end for
18:  end for
19: Return  $M_c, M_r, \text{prec}$ 

```

safer to work with lattices given by vectors defined over  $\mathbb{Z}$ , hence the introduction of  $\hat{A}$ . The integer **prec** corresponds to a truncation precision that will allow us to work over  $\mathbb{Z}^N$  and keep enough information from  $A$  at the same time.

*Remark 5.4.* By construction,  $|\hat{A}[i, j]| \leq |A[i, j]|$  for all  $i, j$ . Hence, Theorem 5.2 and its corollaries, which proceed by upper bounding the coefficients of  $A$  and applying Theorem 5.1, also hold for  $(\det \hat{A}^t \hat{A})^{1/2}$ .

The rows of  $\hat{A}$  generate the lattice that will be reduced in our algorithm.

**Lemma 5.5.** *The  $\mathbb{Z}$ -module generated by the rows of  $\hat{A}$  is a lattice of rank  $N$ .*

*Proof.* For  $i = 0, \dots, N - 1$ ,  $2^{\text{prec}} A_2[i, i] \geq 2^5 N$ , hence  $\hat{A}_2[i, i] \geq 2^{5-\text{prec}} N > 0$ . This shows that the matrix  $\hat{A}_2$  is an invertible diagonal matrix, so that the matrix  $\hat{A}$  has full rank  $N$ .  $\square$

The matrices  $M_c$  and  $M_r$  computed in Algorithm 1 correspond to the scaled matrices  $2^{\text{prec}} \hat{A}_1$  and  $2^{\text{prec}} \hat{A}_2$ .

**Algorithm 2** 1D approach to Problem 2.6

**Input:** Two real numbers  $a < b$ ,  $f$  a transcendental function analytic in a complex neighbourhood of  $[a, b]$ , three nonzero integers  $d, u, v$ , two real numbers  $\rho > 1, \omega_0$  such that  $1 \leq \rho^{\omega_0} < \rho^{N-1}(\rho - 1)/4$ , where  $N = (d + 1)(d + 2)/2$ .

**Output:** If successful, return  $B \in \mathbb{R}_{>0}$  and a list  $\mathcal{L}$  such that  $\mathcal{L} \supset \{X \in \mathbb{Z} \text{ such that } a \leq X/u \leq b \text{ and there exists } Y \in \mathbb{Z}, |f(\frac{X}{u}) - \frac{Y}{v}| < 1/(2B)\}$ .  $B$  is guaranteed to be at most  $\frac{d\rho^{N-\omega_0-1}(\rho-1)}{4M_{\rho,a,b}(f)}$ .

```

1:  $(M_c, M_r, \text{prec}) \leftarrow$  Algorithm 1  $(a, b, f, d, u, v, \rho, \omega_0)$ ,  $N \leftarrow (d + 1)(d + 2)/2$ 
2:  $M_{LLL}, U \leftarrow$  LLL-reduce the rows of  $(M_c \mid M_r)$  // We assume that our LLL call returns both the reduced basis and the change-of-basis matrix denoted  $U$ .
3: if  $\max(\|(M_{LLL}[0, j])_{0 \leq j \leq 2N-1}\|_2, \|(M_{LLL}[1, j])_{0 \leq j \leq 2N-1}\|_2) \leq 2^{\text{prec}}/(2N)$ 
   then
4:    $L_m \leftarrow [X_1^k X_2^\ell \text{ for } k = 0 \text{ to } d - \ell \text{ for } \ell = 0 \text{ to } d]$ 
      // List of monomials, ordered in a way compatible with Algorithm 1, steps 8–10.
5:    $P_0 \leftarrow \sum_{j=0}^{N-1} U[0, j] L_m[j]$ ,  $P_1 \leftarrow \sum_{j=0}^{N-1} U[1, j] L_m[j]$ ,
6:    $R(X_1) \leftarrow \text{Res}_{X_2}(P_0(X_1, X_2), P_1(X_1, X_2))$ 
7:   if  $R(X_1) \neq 0$  then
8:      $\mathcal{L} \leftarrow \{t \in \mathbb{Z}; R(t) = 0\}$ 
9:      $(B_0, B_1) \leftarrow (0, 0)$ 
      // Monomials in  $X_1$  do not contribute to  $B$ ; given the ordering of  $L_m$  this is obtained by starting the loop at  $k = d + 1$ .
10:    for  $k = d + 1$  to  $N - 1$  do
11:       $\varphi \leftarrow \frac{dL_m[k]}{dX_2}(X_1 = x, X_2 = (1 + 1/v)f(x))$ ,  $J \leftarrow |\varphi|([a, b])$ 
12:       $B_0 \leftarrow B_0 + |U[0, k]| \max(J)$ 
13:       $B_1 \leftarrow B_1 + |U[1, k]| \max(J)$ 
14:    end for
15:    return  $B = \max(B_0, B_1)$ ,  $\mathcal{L}$ 
16:   else
17:     return “FAIL”
18:   end if
19: else
20:   return “FAIL”
21: end if

```

We now derive an explicit expression for  $\text{prec}$ . In order to do that, we henceforth assume that the set  $u[a, b]$ , resp.  $v f([a, b])$ , contains at least one nonzero integer  $n_x$ , resp.  $n_f$ ; note that this assumption is made without loss of generality with respect to our problem, since if the assumption does not hold the problem is trivial.

**Lemma 5.6.** *We have*

$$\text{prec} = \lceil (N - \omega_0 - 1) \log_2(\rho) + \log_2(\rho - 1) + \log_2(N) \rceil + 3$$

and

$$8N\rho^{N-\omega_0-1}(\rho - 1) \leq 2^{\text{prec}} \leq 16N\rho^{N-\omega_0-1}(\rho - 1).$$

*Proof.* From our assumption, we have  $uM_{\rho,a,b}(x) \geq u|n_x| \geq 1$  and  $vM_{\rho,a,b}(f) \geq v|n_f| \geq 1$ . It then follows  $R_{2,i} \geq 4/(\rho^{N-\omega_0-1}(\rho - 1)) = R_{2,0}$  for all  $i$ . Therefore, we get  $\text{prec} = \lceil -\log_2(R_{2,0}) + \log_2(N) \rceil + 3$ .  $\square$

*Remark 5.7.* Note that the proof shows in particular that  $R_{2,i} \geq R_{2,0}$  for all  $i$ . If  $\rho^{\omega_0} R_{2,0} \geq 1$ , we thus have  $\rho^{\omega_0} R_{2,i} \geq 1$  for all  $i$  and the determinant  $\det(A^t A)^{1/2}$  is larger than 1, meaning that the method fails as claimed at the beginning of this section. This explains the condition  $\rho^{\omega_0} < \rho^{N-1}(\rho - 1)/4$ .

5.2.1. *Uniformly small polynomials over a transcendental analytic curve.* This part is devoted to the proof of the following.

**Theorem 5.8.** *Let  $d, u, v$  be nonzero integers,  $N = (d+1)(d+2)/2$ ,  $(f_j)_{0 \leq j \leq N-1} = (u^k x^k v^\ell f(x)^\ell)_{\substack{0 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}}$ . Let  $\omega_0 \geq 0$ ,  $\rho > 1$  such that there exists  $\Lambda = (\lambda_{k,\ell})_{\substack{0 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}} \in \mathbb{Z}^N$  with  $\|\Lambda \hat{A}\|_2 \leq 1/(2N)$ , and let  $P(X_1, X_2) = \sum_{0 \leq k+\ell \leq d} \lambda_{k,\ell} X_1^k X_2^\ell$ .*

*Then, we have*

$$(1) \sup_{x \in [a,b]} |P(ux, vf(x))| < 1/2;$$

$$(2) \sup_{\substack{x \in [a,b], z \in f([a,b]) \\ 0 < |y-z| < 1/(2d)}} \left| \frac{P(ux, vz) - P(ux, vy)}{z-y} \right| < \frac{d\rho^{N-\omega_0-1}(\rho-1)}{4M_{\rho,a,b}(f)}.$$

*Proof.* The assumption  $\|\Lambda \hat{A}\|_2 \leq 1/(2N)$  implies in particular, for all  $k, \ell$ ,

$$1/(2N) \geq |\lambda_{k,\ell}| \min_i \hat{A}_2[i, i] \geq 2^{5-\text{prec}} N |\lambda_{k,\ell}|,$$

cf. proof of Lemma 5.5, hence  $|\lambda_{k,\ell}| \leq 2^{\text{prec}-6}/N^2$ . For  $j = 0, \dots, 2N-1$ , we have  $|(\Lambda A)[j] - (\Lambda \hat{A})[j]| \leq \sum_{0 \leq k+\ell \leq d} |\lambda_{k,\ell}| 2^{-\text{prec}} \leq 2^{-6}/N$ , from which follows  $\|\Lambda A\|_1 \leq \|\Lambda \hat{A}\|_1 + 2^{-5}$ .

Let  $P$  be as in the statement of the Theorem, and  $Q(x) = \sum_{j=0}^{N-1} q_j T_{j,[a,b]}(x)$  be the interpolation polynomial of  $P(ux, vf(x))$  at the order  $N$  Chebyshev nodes of the first kind. Then, the coordinates of  $\Lambda A_1$  are exactly  $q_j$ ,  $0 \leq j \leq N-1$ . Proposition 4.1 shows that

$$\begin{aligned} \sup_{x \in [a,b]} |Q(x) - P(ux, vf(x))| &\leq 4 \frac{M_{\rho,a,b}(P)}{\rho^{N-1}(\rho-1)} \\ &\leq 4 \sum_{0 \leq k+\ell \leq d} |\lambda_{k,\ell}| \frac{u^k M_{\rho,a,b}(x)^k v^\ell M_{\rho,a,b}(f)^\ell}{\rho^{N-1}(\rho-1)}, \end{aligned}$$

hence

$$\begin{aligned} \sup_{x \in [a,b]} |P(ux, vf(x))| &\leq \sup_{x \in [a,b]} |Q(x)| + 4 \sum_{0 \leq k+\ell \leq d} |\lambda_{k,\ell}| \frac{u^k M_{\rho,a,b}(x)^k v^\ell M_{\rho,a,b}(f)^\ell}{\rho^{N-1}(\rho-1)}, \\ (5.5) \quad &\leq \sum_{0 \leq j \leq N-1} |q_j| + 4\rho^{\omega_0} \sum_{0 \leq k+\ell \leq d} |\lambda_{k,\ell}| \frac{u^k M_{\rho,a,b}(x)^k v^\ell M_{\rho,a,b}(f)^\ell}{\rho^{N-1}(\rho-1)} \\ &= \|\Lambda A\|_1 \leq \|\Lambda \hat{A}\|_1 + 2^{-5} \\ &\leq 2^{-5} + \sqrt{2N} \|\Lambda \hat{A}\|_2 \text{ thanks to Cauchy-Schwarz inequality} \\ (5.6) \quad &\leq 2^{-5} + 1/\sqrt{2N} < 1/2 \text{ since } N \geq 3. \end{aligned}$$

Finally, for  $x \in [a, b]$ ,  $0 < |y - f(x)| < 1/(2d)$ , we have

$$\begin{aligned}
\left| \frac{P(ux, vy) - P(ux, vf(x))}{y - f(x)} \right| &\leq \sum_{\substack{1 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}} \ell |\lambda_{k,\ell}| u^k |x|^k v^\ell \max(|f(x)|, |y|)^{\ell-1} \\
&\leq \sum_{\substack{1 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}} \ell |\lambda_{k,\ell}| u^k |x|^k v^\ell |f(x)|^{\ell-1} \underbrace{\left(1 + \frac{1}{2d}\right)^{\ell-1}}_{\leq 2} \\
&\leq 2d \sum_{\substack{1 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}} |\lambda_{k,\ell}| u^k M_{\rho,a,b}(x)^k v^\ell M_{\rho,a,b}(f)^{\ell-1} \\
&\leq \frac{d\rho^{N-\omega_0-1}(\rho-1)}{4M_{\rho,a,b}(f)}
\end{aligned}$$

thanks to the comparison of (5.5) to (5.6).  $\square$

From the second point of Theorem 5.8, we can deduce a lower bound for  $|Y/v - f(X/u)|$  in the following way:

**Corollary 5.9.** *With the notations and assumptions of Theorem 5.8, we have, for all  $X \in \mathbb{Z}$ ,  $a \leq X/u \leq b$ , all  $Y \in \mathbb{Z}$ , either:*

$$\left| \frac{Y}{v} - f\left(\frac{X}{u}\right) \right| \geq \frac{2M_{\rho,a,b}(f)}{d\rho^{N-\omega_0-1}(\rho-1)} \text{ or } P(X, Y) = 0.$$

*Proof.* We have, for any  $x \in [a, b]$ ,  $y \in \mathbb{R}$ ,

$$(5.7) \quad P(ux, vy) = P(ux, vf(x)) + (y - f(x)) \frac{P(ux, vy) - P(ux, vf(x))}{y - f(x)}.$$

As  $P \in \mathbb{Z}[X_1, X_2]$  and  $X, Y \in \mathbb{Z}$ , we must have  $P(X, Y) \in \mathbb{Z}$ , so that either  $P(X, Y) = 0$  or  $|P(X, Y)| \geq 1$ . In the former case, there is nothing to prove. In the latter case, either  $|Y/v - f(X/u)| \geq 1/(2d)$  and again there is nothing to prove, or, by plugging  $x = X/u$  and  $y = Y/v$  into (5.7), we obtain

$$\left| \frac{Y}{v} - f\left(\frac{X}{u}\right) \right| \geq \frac{1/2}{\sup_{x \in [a, b], 0 < |y - f(x)| < \frac{1}{2d}} \left| \frac{P(ux, vy) - P(ux, vf(x))}{y - f(x)} \right|}.$$

Overall, we thus have

$$\left| \frac{Y}{v} - f\left(\frac{X}{u}\right) \right| \geq \min\left(\frac{1}{2d}, \frac{2M_{\rho,a,b}(f)}{d\rho^{N-\omega_0-1}(\rho-1)}\right)$$

or  $P(X, Y) = 0$ . But the proof of Lemma 5.6 actually shows that  $\frac{4vM_{\rho,a,b}(f)}{\rho^{N-\omega_0-1}(\rho-1)} \leq 1$ . Hence,

$$\frac{d\rho^{N-\omega_0-1}(\rho-1)}{4M_{\rho,a,b}(f)} \geq dv \geq d,$$

which concludes the proof.  $\square$

**5.3. Proof of the algorithms.** We now have all the elements to prove Algorithm 2. First, recall that the matrices  $M_c$  and  $M_r$  computed in Algorithm 1 correspond to the scaled matrices  $2^{\text{prec}}\hat{A}_1$  and  $2^{\text{prec}}\hat{A}_2$ . If the condition stated at Step 3 of Algorithm 2 is satisfied, then Theorem 5.8 and Corollary 5.9 prove that, given an integer pair  $(X, Y)$ , either  $P_0(X, Y) = P_1(X, Y) = 0$ , or the lower bound of Corollary 5.9 holds.

Steps 6 & 8 of Algorithm 2 deal with the former case, by trying to solve the polynomial system  $P_0(x, y) = P_1(x, y) = 0$ . If  $P_0$  and  $P_1$  are not coprime, however, this will fail ; this is what makes our algorithm (and actually all bivariate versions of Coppersmith's method) heuristic. When this situation occurs, one convenient solution is to subdivide the interval of interest into two subintervals and to apply the algorithm again to these subintervals.

**5.4. Practical remarks.** Algorithm 1 has been written with readability in mind. We now add some practical clarifications. We will discuss some experiments in Section 8.

**5.4.1. Efficiency.** The number of DCT calls (see Step 12 in Algorithm 1) can be reduced from  $O(N)$  to  $O(d)$  by noticing that the DCT  $\delta'$  of the vector  $u'$  associated to  $x\varphi(x)$  can be deduced from the DCT  $\delta$  of the vector  $u$  associated to  $\varphi(x)$  via the following formulas, which are easily deduced from the recurrence relation  $2xT_n(x) = T_{n+1}(x) + T_{n-1}(x)$ :

- $\delta'[0] = (b - a)\delta[1]/4 + (b + a)\delta[0]/2$ ,
- $\delta'[1] = (b - a)\delta[0]/2 + (b - a)\delta[2]/4 + (b + a)\delta[1]/2$ ,
- $\delta'[k] = (b - a)(\delta[k - 1] + \delta[k + 1])/4 + (b + a)\delta[k]/2$ ,  $2 \leq k \leq n - 2$ ,
- $\delta'[n - 1] = (b - a)\delta[n - 2]/4 + (b + a)\delta[n - 1]/2$ .

This gives, in practice, a significant speedup in the construction of the matrix for large  $d$ .

A similar strategy applies for the computation of the remainder matrix  $M_r$ .

**5.4.2. Overestimation issues.** We now discuss the instruction  $B_f \leftarrow \max(g([0, 2\pi]))$ , presented at Step 6 of Algorithm 1. For some functions such as  $\exp$  or  $\Gamma$ , we can take advantage of a closed expression for this maximum. Otherwise, either we develop a dedicated routine to derive a tight estimate of this value, or we can use interval or ball arithmetic [66, 31] to quickly obtain an upper bound, which then may raise overestimation issues. So far, our experiments, which use Arb<sup>11</sup>, resp. MPFI<sup>12</sup>, for every ball, resp. interval, arithmetic based computation, did not show any problematic overestimation that forced us to drop the quick second option.

Also note that since the computed values in  $M_r$  are larger than the actual ones, if the condition stated at Step 3 of Algorithm 2 is satisfied with these computed values, then this condition is also satisfied for the actual values and Theorem 5.8 and Corollary 5.9 apply.

**5.4.3. Rounding issues.** The roundings that occur at Steps 7, 14 and 16 need not be correct roundings ; a satisfactory trade-off is to secure that the values computed, before rounding, at these steps lie at a distance at most  $1/2$  of the corresponding exact mathematical value, which is achieved by using a slightly larger precision in the intermediate computations.

<sup>11</sup><https://arblib.org/>

<sup>12</sup><https://gitlab.inria.fr/mpfi>

We have to determine this value of the intermediate precision: to determine it, we shall add the binary logarithm of the absolute value of the computed number (or an upper bound for it) and the number of arithmetical operations involved in the computation.

Hence, in practice, we work with a precision  $\text{prec}'$ , instead of  $\text{prec}$ . The former precision is obtained from Step 7, provided that the mathematical value  $(N - 1) \log_2(\rho) + \log_2(\rho - 1) + \log_2(N) - \omega_0 \log_2(\rho)$  lies at a distance at most  $1/2$  of the computed value. This is ensured as soon as we work with an internal precision at least equal to  $\max((N - 1) \log_2(\rho), \log_2(\rho - 1), \log_2(N), \omega_0 \log_2(\rho)) + 3$ . Note that  $\text{prec}' \in \{\text{prec}, \text{prec} + 1\}$ .

Then, we replace  $\text{prec}$  with  $\text{prec}'$  in the sequel of Algorithm 1 and in Algorithm 2. This implies, for  $i = 0, \dots, N - 1, j = 0, \dots, N - 1$ ,

$$(5.8) \quad |M_c[i, j]| \leq 2^{\text{prec}' + 1} \max(u|a|, u|b|, v \max_{x \in [a, b]} |f(x)|)^d,$$

$$(5.9) \quad |M_r[i, i]| \leq \frac{2^{\text{prec}' + 1}}{\rho^{N - \omega_0 - 1}(\rho - 1)} \max(u(a + b + \rho(b - a)), vM_{\rho, a, b}(f))^d \\ \leq 8N \max(u(a + b + \rho(b - a)), vM_{\rho, a, b}(f))^d \text{ from Lemma 5.6.}$$

We now have to work with  $M'_c$ , instead of  $M_c$ . The former matrix is obtained from Step 14, provided that the mathematical values  $2^{\text{prec}'} L_{\text{DCT}}[j]$ 's lie at a distance at most  $1/2$  of the corresponding computed value. We deduce from (5.8) that this is ensured as soon as we work with an internal precision at least equal to  $\text{prec}' + 2N + 1 + d \log_2(\max(u|a|, u|b|, v \max_{x \in [a, b]} |f(x)|))$ . Note that  $M_c[i, j] - M'_c[i, j] \in \{0, \text{sgn}(M_c[i, j])\}$ .

Likewise, we now have to work with  $M'_r$ , instead of  $M_r$ . The former matrix is obtained from Step 16, provided that the mathematical values  $2^{\text{prec}'} \frac{4(uB_x)^k (vB_f)^\ell}{\rho^{N - \omega_0 - 1}(\rho - 1)}$ 's lie at a distance at most  $1/2$  of the corresponding computed value. We deduce from (5.9) that this is ensured as soon as we work with an internal precision at least equal to  $\log_2(N) + 2N + 4 + d \log_2(\max(u(a + b + \rho(b - a)), vM_{\rho, a, b}(f)))$ . Note that  $M_r[i, j] - M'_r[i, j] \in \{0, 1\}$ .

If we denote  $\hat{A}' = (M'_c M'_r)$ , we notice that since  $|\hat{A}'[i, j]| \leq |A[i, j]|$  for all  $i, j$ , Remark 5.4 applies again: Theorem 5.2 and its corollaries hold for  $\hat{A}'$ .

We now analyse what are the consequences for the proof of Theorem 5.8. We have  $|\lambda_{k, \ell}| \leq 2^{\text{prec} - 5} / N^2$  for any  $k, \ell$ . If we denote  $\hat{A}' = (M'_c M'_r)$ , we obtain for  $j = 0, \dots, 2N - 1$ ,

$$|(\Lambda A)[j] - (\Lambda \hat{A}')[j]| \leq \sum_{0 \leq k + \ell \leq d} |\lambda_{k, \ell}| 2^{1 - \text{prec}'} \leq 2^{-4} / N,$$

from which follows  $\|\Lambda A\|_1 \leq \|\Lambda \hat{A}'\|_1 + 2^{-4}$ . The upper bound in Inequality 5.6 becomes  $2^{-4} + 1/\sqrt{2N} < 1/2$  since  $N \geq 3$ .

**5.4.4. Newton polynomials.** In practice, we replace the monomial functions  $\{u^k x^k\}_{0 \leq k \leq d}$  with Newton polynomials functions  $\{ux(ux - 1) \cdots (ux - k + 1)/k!\}_{0 \leq k \leq d}$ . In both cases, the substitution  $x = X/u$  yields integer values - respectively  $\{X^k\}_{0 \leq k \leq d}$  and  $\{X(X - 1) \cdots (X - k + 1)/k!\}_{0 \leq k \leq d}$ . Likewise, we replace the ‘‘monomials’’  $\{v^k f(x)^k\}_{0 \leq k \leq d}$  with ‘‘Newton polynomials’’  $\{vf(x)(vf(x) - 1) \cdots (vf(x) - k + 1)/k!\}_{0 \leq k \leq d}$ . Hence, the changes to operate are:

- Step 10, Algorithm 1,  

$$\varphi \leftarrow \left( x \mapsto \prod_{j=1}^k (ux - j + 1)/j \prod_{j=1}^{\ell} (vf(x) - j + 1)/j \right),$$
- Step 16, Algorithm 1,  

$$M_r[i, i] \leftarrow \left\lfloor 2^{\text{prec}} \frac{4}{\rho^{N-\omega_0-1}(\rho-1)} \prod_{j=1}^k \frac{uB_x+j-1}{j} \prod_{j=1}^{\ell} \frac{vB_f+j-1}{j} \right\rfloor,$$
- Step 4, Algorithm 2,  

$$L_m \leftarrow \left[ \prod_{j=1}^k \frac{X_1-j+1}{j} \prod_{j=1}^{\ell} \frac{X_2-j+1}{j} \text{ for } k = 0 \text{ to } d - \ell \text{ for } \ell = 0 \text{ to } d \right].$$

The use of Newton polynomials leads to smaller uniform norms, hence makes it possible to tackle larger intervals for a same  $d$ . This improvement is asymptotically negligible (it contributes to a lower order term) but is quite significant in practice.

Note that the iterations described in Section 5.4.1 can easily be adapted to deal with Newton polynomials.

**5.5. Analysis.** If we apply the LLL lattice basis reduction algorithm to  $\hat{A}$ , we obtain:

**Corollary 5.10.** *Assume that  $\det(\hat{A}^t \hat{A})^{1/2(N-1)} \leq \frac{2^{-(N+3)/4 - \text{prec}/(N-1)}}{N}$ ; then Theorem 5.8 applies with any of the first two vectors of an LLL-reduced basis of the lattice generated by the rows of  $\hat{A}$ .*

*Proof.* Let  $A' = 2^{\text{prec}} \hat{A} \in \mathcal{M}_n(\mathbb{Z})$ , we know from Theorem 4.4 that if  $w_1$  and  $w_2$  denote these first two vectors, we have

$$\|2^{\text{prec}} w_i\|_2 \leq 2^{(N-1)/4} \max(\det(A'^t A')^{1/2(N-1)}, \det(A'^t A')^{1/(2N)})$$

from which follows

$$2^{\text{prec}} \|w_i\|_2 \leq 2^{(N-1)/4} 2^{\text{prec}} \max(2^{\text{prec}/(N-1)} \det(\hat{A}^t \hat{A})^{1/2(N-1)}, \det(\hat{A}^t \hat{A})^{1/(2N)}),$$

hence

$$\|w_i\|_2 \leq 2^{(N-1)/4} \max(2^{\text{prec}/(N-1)} \det(\hat{A}^t \hat{A})^{1/2(N-1)}, \det(\hat{A}^t \hat{A})^{1/(2N)}) \leq \frac{1}{2N}.$$

□

We shall base our analysis on Inequality (5.4). The important term in the analysis is  $(uv)^{2N/(3(d+3))} M_{\rho,a,b}(f)^{2N/(3(d+3))} / \rho^{N/2}$ . The quality of the bound thus depends on the size of  $\rho$  and the growth of  $f$ .

We shall start by giving a general result. This result will then be turned into more readable versions under various sets of assumptions in Theorems 5.16, 5.23 and 5.24.

**Proposition 5.11.** *Let  $f$  be analytic in a neighbourhood of the closed disc  $\mathcal{D}_{a,b,K} = \{z \in \mathbb{C} : |z - (a+b)/2| \leq K/2\}$ ,  $d$  be an integer  $\geq 2$ ,  $N = (d+1)(d+2)/2$ , and  $\omega_0 \geq 0$ ,  $\rho = K/(b-a) \geq 2$  be two real parameters. Let  $M_{\mathcal{D}_{a,b,K}}(f) := \sup_{z \in \mathcal{D}_{a,b,K}} |f(z)|$ .*

*Then, if*

$$b - a < K \left( 2^{6(d+3)} uv (|a+b| + K) M_{\mathcal{D}_{a,b,K}}(f) \right)^{-\frac{2dN}{3(N(N-3)+2\omega_0 + \lceil \omega_0 \rceil (\lceil \omega_0 \rceil - 2\omega_0 + 1))}},$$

*we have  $\Delta_{N,[a,b],\omega_0}^{1/(N-1)} < \frac{2^{-(N+3)/4 - \text{prec}/(N-1)}}{N}$ .*



*Proof.* We first notice that under our assumption  $\rho = K/(b - a) \geq 2$ , we have  $E_{\rho,a,b} \subset \mathcal{D}_{a,b,K}$ . Thanks to Corollary 5.3, in view of  $(\rho/(\rho - 1))^{N/(N-1)} \leq 2^{3/2}$ , we have

$$\Delta_{N,[a,b],\omega_0}^{1/(N-1)} \leq 36\sqrt{2N} \frac{(uv(|a+b|+K)/2)^{2N/(3(d+3))}}{\rho^{N/2+\lfloor\omega_0\rfloor(\lfloor\omega_0\rfloor-2\omega_0+1)/(2(N-1))}} M_{\mathcal{D}_{a,b,K}}(f)^{2N/(3(d+3))}.$$

Thus, for  $\Delta_{N,[a,b],\omega_0}^{1/(N-1)} < 2^{-(N+3)/4} 2^{-\text{prec}/(N-1)}/N$ , it suffices, using Lemma 5.6 that

$$\rho > \left( 36 \cdot 2^{(N+5)/4 - 2N/(3(d+3)) + 4/(N-1)} N^{3/2} \right. \\ \left. (uv(|a+b|+K)M_{\mathcal{D}_{a,b,K}}(f))^{2N/(3(d+3))} \right)^{\frac{2(N-1)}{N(N-3)+2\omega_0+\lfloor\omega_0\rfloor(\lfloor\omega_0\rfloor-2\omega_0+1)}}.$$

We observe that

$$36 \cdot 2^{(N+5)/4 - 2N/(3(d+3)) + 3/(N-1)} N^{3/2} < 2^{4N}$$

for  $d \geq 1$ , from which the proposition follows.  $\square$

**Corollary 5.12.** *Under the assumptions of Proposition 5.11, Algorithm 2 over  $[a, b]$  produces at Step 5 two polynomials  $P_0, P_1$  such that  $\sum_{x \in [a,b]} |P_i(ux, vf(x))| < 1/2$  for  $i \in \{0, 1\}$ . In particular, Algorithm 2 never reaches Step 20 and its output is valid.*

*Proof.* It suffices to apply Proposition 5.11, Corollary 5.10, Theorem 5.8 and Corollary 5.9.  $\square$

Note that the algorithm may still return “FAIL” at Step 17, precisely in the case where  $P_0$  and  $P_1$  are not coprime – this makes the algorithm heuristic.

**5.6. Complexity analysis.** In this subsection, we deduce from the bounds estimates for the complexity of our algorithm applied to a fixed interval  $[\alpha, \beta]$ . This actually requires several things:

- An evaluation of the complexity of the basic blocks, namely Algorithms 1 and 2.
- Use Corollary 5.12 to evaluate the size of a subinterval  $[a, b]$  which can be treated at once by those algorithms ;
- Investigate the interplay between  $d$  and  $\omega_0$ , two parameters which have an impact on both the complexity and the quality of the final bound on  $1/w$ .

We shall split the discussion into two parts;  $\rho = K/(b - a)$  for fixed  $K$ , and  $\rho = K/(b - a)$  where  $K$  may depend on  $d$  (and  $d \rightarrow \infty$ ).

The first part corresponds to the practical situation of a fixed degree, and applies to any function which is sufficiently regular around the interval  $[a, b]$  under study.

The second part, on the other hand, contains theoretical results showing that a slightly better complexity can be achieved for entire functions.

We start by giving complexity estimates for Algorithms 1 and 2.

**5.6.1. Complexity of Algorithms 1 and 2.** In this subsection, we denote  $M(n)$  the complexity of multiplying two  $n$ -bits integers (or two precision  $n$  floating points numbers).

We assume that interval evaluation at precision  $p$  of a function  $f$  uses  $O(1)$  evaluations of  $f$  at precision  $p$ . This is not merely a convenient theoretical assumption:

efficient interval libraries are available; in our implementation, we used the Arb library [31].

**Lemma 5.13.** *Put  $\mathcal{M} = \max(2, u, v, |a|, |b|, \rho, M_{\rho,a,b}(f))$ . The computations of Algorithm 1 on input  $a, b, f, d, u, v, \rho, \omega_0$  can be made in floating-point precision  $\mathfrak{p} = O(\text{prec} + d \log \mathcal{M})$ .*

*In particular, for fixed  $a, b, \rho, \omega_0, f$ , for  $u, v = 2^p$ , with  $p \geq d$ , the required precision is  $O(dp)$ .*

*Proof.* We need to be able to round, cf. Section 5.4.3, at Steps 14 and 16; we need a precision  $\text{prec}$  (or  $\text{prec} + 1$ , as discussed in Section 5.4.3) after the binary point, for real numbers which are at most  $O(\max(u, v)^d (\max(|a|, |b|)\rho)^d M_{\rho,a,b}(f)^d)$ .  $\square$

**Proposition 5.14.** *On input  $a, b, f, d, u, v, \rho, \omega_0$ , assuming that evaluating  $f$  in precision  $p$  costs  $C_{f,p}$ , and a DCT of size  $n$  in precision  $q$  has cost  $\tilde{O}(nM(q))$ , Algorithm 1 has complexity*

$$\tilde{O}(d^4 M(\mathfrak{p}) + d^2 C_{f,\mathfrak{p}}),$$

where  $\mathfrak{p}$  is as in the previous Lemma.

*Proof.* The most costly steps of Algorithm 1 appear in the loop 8-18 which can be performed using  $O(d^2)$  evaluations of  $f$  at precision  $O(\text{prec})$ , and  $O(d^4)$  multiplications of real numbers in precision  $O(\text{prec})$ , plus  $O(d^2)$  DCT of size  $N$  in precision  $\text{prec}$ .  $\square$

In particular, for fixed  $a, b, \rho$ ,  $u = v = 2^p$ ,  $p \geq d$ ,  $C_{f,\text{prec}} = \tilde{O}(M(\text{prec}))$ ,  $C_{DCT}(n) = \tilde{O}(M(n))$ , and assuming finally  $M(n) = \tilde{O}(n^\mu)$ , and ignoring the  $\omega_0 \log \rho$  term in  $\text{prec}$ , we obtain a complexity of  $\tilde{O}(d^{4+\mu} p^\mu)$ .

We now turn to the analysis of Algorithm 2. We shall limit ourselves to the analysis of Steps 1-5, which compute the two auxiliary polynomials. This is, in any case, the core of the algorithm, but also the choice made in previous papers, and thus allows for a better comparison.

**Proposition 5.15.** *On input  $a, b, f, d, u, v, \rho, \omega_0$ , under the assumption  $C_{f,\mathfrak{p}} = O(\mathfrak{p}^2)$  and  $C_{DCT}(n) = O(n^2)$ , Steps 1-5 of Algorithm 2 have complexity  $O(d^6 M(d^2) \mathfrak{p}^2)$ .*

*Proof.* The main steps of Algorithm 2 are:

- A call to Algorithm 1 ;
- A call to LLL on a lattice of dimension  $N$  with entries of size  $O(\mathfrak{p})$ .

For the second part, we use the  $L^2$  algorithm [52] on a lattice of dimension  $N = O(d^2)$ , embedded into  $\mathbb{R}^{2N}$ ; we thus have complexity  $O(d^6 M(d^2) \mathfrak{p}^2)$  (note that  $\mathfrak{p} \geq d$ ). This cost dominates the cost of Algorithm 2.  $\square$

5.6.2. *Number of subintervals for fixed  $d$ .* Thanks to the results of the previous subsection, we can estimate the maximum size of an interval  $[a, b] \subset [\alpha, \beta]$ , with  $\alpha, \beta$  fixed, for which Algorithm 2 succeeds (in the sense of Corollary 5.12). This follows from Proposition 5.11, and yields at the same time the number of subintervals to be considered if one wants to deal with a full interval  $[\alpha, \beta]$

**Theorem 5.16.** *Given fixed  $f$ , a fixed parameter  $d$ , two fixed real numbers  $\alpha$  and  $\beta$ , Problem 2.6 can heuristically be solved for  $u, v \rightarrow \infty$  over  $[\alpha, \beta]$  using*

$$(5.10) \quad O\left((\beta - \alpha)(uv)^{\frac{2Nd}{3(N(N-3)+2\omega_0+\lfloor\omega_0\rfloor(\lfloor\omega_0\rfloor-2\omega_0+1))}}\right)$$

calls to Algorithm 2 with parameter  $d$ .

We then obtain a value

$$(5.11) \quad w = O\left((uv)^{\frac{2N(N-\omega_0)d}{3(N(N-3)+2\omega_0+\lfloor\omega_0\rfloor(\lfloor\omega_0\rfloor-2\omega_0+1))}}\right).$$

*Proof.* This is a direct consequence of Corollary 5.12, where we choose  $K = 2(b - a)$ . The heuristic nature of this result comes from the possibility that the two polynomials obtained in Algorithm 2 are not coprime, in which case one cannot recover the solutions  $X, Y$  from those two polynomials.

Finally, thanks to Corollary 5.9, the upper bound is  $O(\rho^{(N-\omega_0)})$ , from which the second part of the result follows.  $\square$

*Remark 5.17.* If one is only interested with the smallest possible complexity, it should be noted that the exponent in (5.10) is minimal for  $\omega_0 \in [1, 2]$ , and equal in this case to  $8N/(3(d+3)(d^2+3d-2))$ ; for  $\omega_0 = 2$  we then get  $w = O((uv)^{4N/(3(d+3))})$ .

*Remark 5.18.* In the case where  $\omega_0$  is an integer, the bounds take the nicer form

$$O\left((\beta - \alpha)(uv)^{\frac{2Nd}{3(N-\omega_0)(N+\omega_0-3)}}\right) \text{ and } w = O\left((uv)^{\frac{2Nd}{3(N+\omega_0-3)}}\right).$$

In particular, this shows the limits of the approach: for  $\omega_0 = N$ , we decrease the exponent of the bound on  $w$  by a factor of 2, but can expect nothing better. We shall see in the next section how to go beyond this limitation.

We now discuss the case where we let  $d \rightarrow \infty$ . This has two goals:

- See for what value of  $d$  we can expect to treat a whole interval  $[\alpha, \beta]$  at once; notice that better results will be obtained later on if  $f$  is entire and we have control on its growth at infinity;
- Give a simplified form of the estimates of Theorem 5.16, which, in reason of the technical parameter  $\omega_0$ , are rather unpleasant and unintuitive.

**Corollary 5.19.** *Let again  $\alpha, \beta$  be fixed real numbers. For  $d \rightarrow \infty$ , for  $\omega_0 = \lambda N(1 + o(1))$ ,  $\lambda \in [0, 1)$ , Problem 2.6 can heuristically be solved for  $u, v \rightarrow \infty$  over  $[\alpha, \beta]$  using*

$$(5.12) \quad O\left((\beta - \alpha)(uv)^{\frac{4(1+o(1))}{3(1-\lambda^2)d}}\right)$$

calls to Algorithm 2 with parameter  $d$ , giving a bound

$$(5.13) \quad w = O\left((uv)^{\frac{2d(1+o(1))}{3(1+\lambda)}}\right).$$

*Remark 5.20.* Note that if we assume  $d = \Theta(\log(uv))$ , Corollary 5.19 states that one call is enough to address an interval of the size  $\Omega(\beta - \alpha)$  and we then obtain  $w = e^{O(\log^2(uv))}$ . We will improve this result in Section 5.6.4 under additional assumptions on the growth of  $f$  at infinity.

*Remark 5.21.* In all this section, our complexity estimates should be considered as somewhat pessimistic for fixed  $d$ , at least for usual transcendental functions. Indeed, we base our complexity estimates on estimates on the size of the *second* vector of an LLL-reduced basis, estimates which can only be obtained under a (trivial, thus extremely pessimistic in practice) lower bound on the size of the first vector.

For a “classical” function such as  $\exp$  or  $\Gamma$ , we notice in practice that most of the time, the second vector has a size similar to the size of the first one. Under this assumption, the bound then drops to  $O((\beta - \alpha)(uv)^{\frac{4}{3(d+3)}})$ , with  $w = O((uv)^{\frac{4N}{3(d+3)}})$ .

Note that it is easy to build examples where this latter bound does not hold, by taking a function which has a very good algebraic approximation (which gives a very short first vector) over the interval under study.

5.6.3. *Tuning  $d$  and  $\omega_0$ .* Again, in order to ease this very technical discussion, we shall focus on the situation of a fixed  $f, a, b$  for  $uv \rightarrow \infty$ .

Let us start by pointing that a tedious, but not difficult computation shows that for  $d \geq 2$  the exponent in (5.11) is decreasing for  $\omega_0 \in [0, N]$ ; the maximal value of this exponent, for  $\omega_0 = 0$ , is  $4dN/(3(d-1)(d+4)) \approx 2d/3$ , whereas its minimal value, for  $\omega_0 \in [N-1, N]$ , is  $dN/(3(d^2 + 3d - 2)) \approx d/3$ . We thus have a *wall-type phenomenon*: if we want to get access to a good complexity, we need to increase  $d$ ; but then  $\omega_0$  fails to prevent the degradation of the estimate on  $w$ , at least in a significant way. In practice, if we target a sharp bound and let  $\omega_0$  grow with this purpose in mind, we observe that the lattice basis reduction step decreases  $d$  to some  $\delta$  on its own simply by not using monomials of degree  $> \delta$  for the first vectors.

We shall however see that setting  $\omega_0$  to a non-zero value still allows one to get a better complexity/ $w$  compromise, and shall study a different method giving complete control on  $w$  in the next section.

From now on, we thus fix a value of  $w = (uv)^\mu$  and try to find a pair  $(d, \omega_0)$  which minimizes the complexity required to achieve this value of  $w$ .

We start with the asymptotic situation, i.e.,  $d \rightarrow \infty$ .

**Proposition 5.22.** *Let  $d \rightarrow \infty$ , and  $\omega_0 = \lambda N(1 + o(1))$ . The value of  $d$  such that the exponent of  $uv$  in (5.13) is  $\mu$  while minimizing the exponent in (5.12) is  $d = 2\mu(1 + o(1))$ , obtained for  $\lambda = 1/3(1 + o(1))$ . This gives a number of subintervals*

$$O\left((\beta - \alpha)(uv)^{\frac{3}{4\mu}(1+o(1))}\right).$$

*Proof.* Elementary calculus. □

Led by this asymptotic statement, we have computed, for small values of  $\mu$ , the degree giving the best estimate for the complexity; it turns out that in all our computations, the optimal value of the degree was  $d = \lfloor 2\mu \rfloor$ , except when  $\mu = r + 1/2$  is an half-integer, where the optimal  $d$  is  $2r$ .

Working out a closed form for the exponent of the complexity estimate as a function of  $\mu$  seems thus possible, but would be moderately enlightening; it seems preferable to give a plot of the corresponding function. Figure 1 gives three curves. The dashed curve corresponds to the best exponent in Theorem 5.16 as a function of the exponent of  $uv$  in the bound on  $w$ . The dotted curve represents a similar function, but using a version of our bounds controlling only the first vector of the lattice. Finally, the plain curve is the asymptotic bound  $3/(4\mu)$ .

5.6.4. *The case  $\rho(b - a) \rightarrow \infty$ .* In this subsection, we shall now let  $K$  depend on  $d$ , namely we shall let it tend to  $\infty$  with  $d$ . We shall thus need the function under study to be an entire function. Recall that if  $f : \mathbb{C} \rightarrow \mathbb{C}$  is an entire function, and if  $\theta = \limsup_{\rho \rightarrow \infty} \log \log \sup_{|z| \leq \rho} |f(z)| / \log \rho$  is finite, the function  $f$  is said to have finite order  $\theta$ .

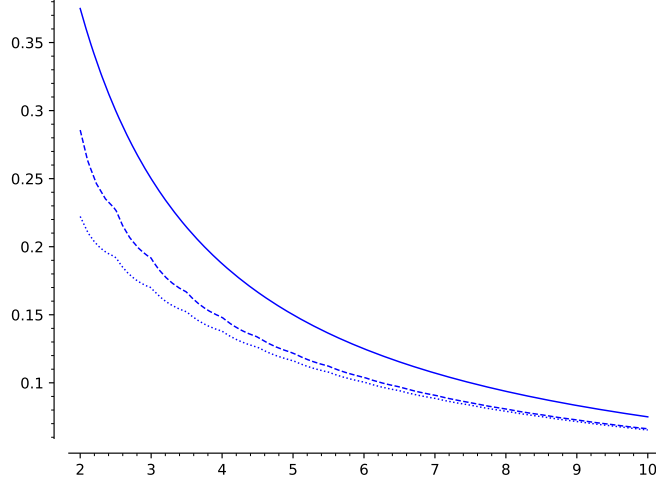


FIGURE 1. Exponent estimates

The presence of a term  $M_{\rho,a,b}(f)$ , depending on the growth of  $f$  at infinity, shows that it is difficult to give a single ready-to-use result. We thus split the discussion into two parts: the case of entire functions of finite order, in which the value of the order gives sufficiently precise information on the growth at infinity to obtain a general result ; and then two examples of a function of order zero and of a function of order infinity – in those cases, case-by-case estimates of the growth at infinity are required.

**Theorem 5.23.** *Let  $f$  be an entire function of finite order  $\leq \theta$ ,  $d \geq 2$  be an integer,  $N = (d+1)(d+2)/2$ , and let  $\omega_0 = \lambda N + o(N)$ , for some fixed  $\lambda \in [0, 1)$ .*

*For  $uv \rightarrow \infty$ , for any constant  $\nu > \frac{4\theta}{3(1-\lambda^2)}$ , for*

$$d = \nu \frac{\log(uv)}{\log \log(uv)} (1 + o(1)),$$

*Algorithm 2 succeeds (in the sense of Corollary 5.12) over  $[a, b]$  and yields a bound on  $w$  of the form:*

$$w \leq (uv)^{\frac{\nu^2(1-\lambda)}{2\theta} \frac{\log(uv)}{\log \log(uv)} (1+o(1))}.$$

*Proof.* Let  $\theta' > \theta$  a parameter which will be fixed later on. We choose  $K = d^{1/\theta'}/2 > 2(b-a)$  for  $d$  large enough. The disc  $\mathcal{D}_{a,b,K}$  is, for our choice of  $\rho$ , included into the ball  $B(0, d^{1/\theta'})$  for  $d$  large enough; hence, the assumption that  $f$  has order  $\theta$  shows that there exists constants  $C_{f,\theta'}, \sigma_{f,\theta'} \in \mathbb{R}$  such that

$$M_{\mathcal{D}_{a,b,K}}(f) \leq C_{f,\theta'} \exp(\sigma_{f,\theta'} d^{\theta'}) = 2^{O(d)}.$$

Proposition 5.11 then implies that a sufficient condition for the conclusion of the theorem to hold is

$$b-a < \frac{1}{4} \left( (|a+b| + d^{1/\theta'}) M_{\mathcal{D}_{a,b,K}}(f) \right)^{-4/(3d)(1+o(1))} d^{1/\theta'} (uv)^{-4/(3d)(1+o(1))}.$$

For  $d$  large enough,  $(|a + b| + d^{1/\theta'})M_{\mathcal{D}_{a,b,K}}(f) > 1$ , so that it suffices to have

$$b - a < \frac{1}{4} \underbrace{\left( (|a + b| + d^{1/\theta'})M_{\mathcal{D}_{a,b,K}}(f) \right)^{-\frac{4}{3d(1-\lambda^2)}(1+o(1))}}_{\text{bounded as } d \rightarrow \infty} d^{1/\theta'}(uv)^{-\frac{4}{3d(1-\lambda^2)}(1+o(1))}.$$

For the condition to hold it thus suffices that the product of the last two terms

$$d^{1/\theta'}(uv)^{-4/(3d(1-\lambda^2))(1+o(1))} \rightarrow \infty,$$

for which it suffices that, for some  $\varepsilon' > 0$ ,

$$d \log d \geq \left( \frac{4\theta'}{3(1-\lambda^2)} + \varepsilon' \right) \log(uv),$$

which obviously holds under the assumption on  $d$  made in the theorem for  $uv$  large enough and  $\theta' < 3(1-\lambda^2)\nu/4$ .

As for the last part, the bound on  $w$  is  $\geq \rho^{N(1-\lambda)} = K^{N(1-\lambda)(1+o(1))}$ , namely we have

$$\log(w) = \frac{(1-\lambda)d^2 \log d}{2\theta'}(1+o(1)) \leq \frac{(1-\lambda)\nu^2 \log^2(uv)}{2\theta \log \log uv}(1+o(1)),$$

as claimed.  $\square$

In particular, for the TMD over  $[1/4, 1/2)$  for the exponential function ( $\theta = 1$ ), hence for  $a = 1/4, b = 1/2$  and  $u = 2^{p+1}$  and  $v = 2^{p-1}$ , for  $p \rightarrow \infty$ , we obtain the condition  $d \geq \left( \frac{8 \log 2}{3(1-\lambda^2)} + \varepsilon \right) \frac{p}{\log p}$  for the full interval  $[1/4, 1/2)$ , with a bound  $w \leq 2^{\left( \frac{32 \log^2 2}{9(1-\lambda^2)(1+\lambda)} + \varepsilon \right) \frac{p^2}{\log p}}$ .

*Two other examples.* We illustrate, more generally, the fact that we get asymptotic results depending on the rate of growth of  $f$  at infinity: the slower the growth of  $f$ , the better the performance.

**Theorem 5.24.** *Let  $f = \exp(\exp(z))$ . For  $uv$  large enough, for any constant  $\nu > \frac{4}{3(1-\lambda^2)}$ , for  $d = \lceil \nu \frac{\log(uv)}{\log \log \log(uv)} \rceil$ , Algorithm 2 succeeds in the sense of Corollary 5.12 and we obtain*

$$w = (uv)^{\nu^2(1-\lambda) \frac{\log(uv) \log \log(uv)}{(\log \log \log(uv))^2} (1+o(1))}.$$

*Let  $g(z) = \sum_{n \geq 0} \exp(-n^2)z^n$ . For  $uv$  large enough, for any constant  $\nu > \frac{4}{3(1-\lambda^2)}$ , for  $d = \lceil \nu \sqrt{\log(uv)} \rceil$ , Algorithm 2 succeeds in the sense of Corollary 5.12 and we obtain  $w = (uv)^{\frac{3}{2}\nu^2(1-\lambda^2)(1-\lambda)\sqrt{\log(uv)}(1+o(1))}$ .*

*Proof.* The proof is similar to the proof of the previous theorem, with  $K = \log d$  for  $f$  and  $K = \exp(3(1-\lambda^2)d/2)$  for  $g$ . See Appendix B.  $\square$

## 6. THE TWO-VARIABLE METHOD

We consider  $u, v \in \mathbb{N} \setminus \{0\}$ ,  $a_1 < b_1$  and  $a_2 < b_2$ , and  $f : [a_1, b_1] \rightarrow \mathbb{R}$  a function that is analytic in a neighbourhood of  $[a_1, b_1]$ . In this section, we develop a heuristic algorithmic approach to determine the integers  $X, Y$  such that

$$(6.1) \quad X/u \in [a_1, b_1] \text{ and } a_2 < f(X/u) - Y/v < b_2.$$

Note that Problem 2.6 is a particular case: let  $w \in \mathbb{N} \setminus \{0\}$ , we set  $[a, b] = [a_1, b_1]$  and  $b_2 = -a_2 = 1/w$ . Without loss of generality, we can assume  $b_2 = -a_2$  by replacing

$f$  by  $f + (a_2 + b_2)/2$ . We prefer to keep  $a_2$  and  $b_2$  arbitrary to put more emphasis on the symmetry between  $(a_1, b_1)$  and  $(a_2, b_2)$  in the formulas and statements.

Our approach aims at building a trap for these pairs  $(X, Y)$ . We compute, by combining two-dimensional Chebyshev interpolation and lattice reduction, two polynomials  $P_0, P_1 \in \mathbb{Z}[X_1, X_2]$  such that, for  $i = 0, 1$ , for all  $x \in [a_1, b_1]$ ,  $t \in [a_2, b_2]$ , we have  $|P_i(ux, v(f(x) + t))| < 1$ . Let  $X \in \mathbb{Z}$  such that  $X/u = x_0 \in [a_1, b_1]$ ,  $Y \in \mathbb{Z}$  such that  $Y/v = f(x_0) + t_0$  with  $t_0 \in [a_2, b_2]$ , we have  $P_i(ux_0, v(f(x_0) + t_0)) = P_i(X, Y) \in \mathbb{Z} \cap (-1, 1) = \{0\}$ , that is to say  $(X, Y)$  is a common root to  $P_0$  and  $P_1$ . As in Section 5, we use our heuristic assumption:  $P_0$  and  $P_1$  are supposed to have no nonconstant common factor. We eliminate one of the variables and get the list of all the integers  $X, Y$  that satisfy (6.1).

In the sequel of this section, we start with estimates of the determinants of the lattices that we use, we present our algorithm, the proof of its correctness and analyse its complexity. When the proofs of the statements are similar to the ones presented in Section 5, we shall postpone them to Appendix D.

*Throughout this section,  $N_1, N_2 \geq 1$  and  $N \geq 2$ , will be three integers. In order to avoid degenerate situations and trivial output, we shall always assume  $N_1 N_2 \geq N$ .*

### 6.1. Volume estimates for rigorous interpolants at the Chebyshev nodes.

We start by introducing the two dimensional extension of the DCT-II:

$$\begin{aligned} \text{2D-DCT-II} : \quad \mathbb{R}^{N_1} \times \mathbb{R}^{N_2} &\rightarrow \mathbb{R}^{N_1} \times \mathbb{R}^{N_2} \\ (x_{\ell_1, \ell_2})_{\substack{0 \leq \ell_1 \leq N_1 - 1 \\ 0 \leq \ell_2 \leq N_2 - 1}} &\mapsto (X_{k_1, k_2})_{\substack{0 \leq k_1 \leq N_1 - 1 \\ 0 \leq k_2 \leq N_2 - 1}} \end{aligned}$$

with

$$X_{k_1, k_2} = \sum_{0 \leq \ell_1 \leq N_1 - 1} \sum_{0 \leq \ell_2 \leq N_2 - 1} x_{\ell_1, \ell_2} \cos\left(\frac{k_1(\ell_1 + 1/2)\pi}{N}\right) \cos\left(\frac{k_2(\ell_2 + 1/2)\pi}{N}\right),$$

for  $k_1 = 0, \dots, N_1 - 1, k_2 = 0, \dots, N_2 - 1$ .

Let  $N \in \mathbb{N}, N \geq 2$ , let  $i = 0, \dots, N - 1$ , let  $f_i$  a function defined over  $[a_1, b_1] \times [a_2, b_2]$ , if we interpolate  $f_i$  by polynomials in  $\mathbb{R}_{N_1 - 1, N_2 - 1}[x, t]$  at pairs of Chebyshev nodes  $\mu_{k_1, k_2} = (\mu_{k_1, N_1 - 1, [a_1, b_1]}, \mu_{k_2, N_2 - 1, [a_2, b_2]})_{\substack{0 \leq k_1 \leq N_1 - 1 \\ 0 \leq k_2 \leq N_2 - 1}}$ , cf. Section 4.1, we have

the following expressions for the interpolation polynomials (the proof is identical to the one variable case [48, Chap. 6]), for  $i = 0, \dots, N - 1$ :

$$Q_i(x, t) = \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} c_{k_1, k_2, i} T_{k_1, [a_1, b_1]}(x) T_{k_2, [a_2, b_2]}(t) \in \mathbb{R}_{N_1 - 1, N_2 - 1}[x, t]$$

with

$$(c_{k_1, k_2, i})_{\substack{0 \leq k_1 \leq N_1 - 1 \\ 0 \leq k_2 \leq N_2 - 1}} = \frac{4}{N_1 N_2} \text{2D-DCT-II} \left( (f_i(\mu_{N_1 - 1 - \ell_1, N_2 - 1 - \ell_2}))_{\substack{0 \leq \ell_1 \leq N_1 - 1 \\ 0 \leq \ell_2 \leq N_2 - 1}} \right).$$

Let  $\rho_1, \rho_2 > 1$ ,  $a_1 < b_1$ ,  $a_2 < b_2$ , we define  $\mathcal{E}_{\rho_1, a_1, b_1, \rho_2, a_2, b_2} = \mathcal{E}_{\rho_1, a_1, b_1} \times \mathcal{E}_{\rho_2, a_2, b_2}$  and  $E_{\rho_1, a_1, b_1, \rho_2, a_2, b_2} = E_{\rho_1, a_1, b_1} \times E_{\rho_2, a_2, b_2}$ . Let  $N_1, N_2 \geq 1$ , and  $N \leq N_1 N_2$ . Let  $f_0, \dots, f_{N-1}$  be functions analytic in a neighbourhood of  $E_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}$ . We

introduce the  $N \times (N_1 N_2 + N)$  matrix  $A = (A_1 A_2)$ , defined by

$$(6.2) \quad \begin{aligned} (A_1)_{i,(k_1,k_2)} &= \left( \frac{c_{k_1,k_2,i}}{2^{\delta_{0k_1} + \delta_{0k_2}}} \right)_{\substack{0 \leq i \leq N-1 \\ 0 \leq k_1 \leq N_1-1, 0 \leq k_2 \leq N_2-1}}, \\ (A_2)_{i,j} &= \delta_{ij} \frac{16\rho_1\rho_2 M_{\rho_1,a_1,b_1,\rho_2,a_2,b_2}(f_i)}{(\rho_1-1)(\rho_2-1)} \\ &\quad \left( \frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}} \right), \quad 0 \leq i, j \leq N-1, \end{aligned}$$

where  $M_{\rho_1,a_1,b_1,\rho_2,a_2,b_2}(g) = \max_{z \in \mathcal{E}_{\rho_1,a_1,b_1,\rho_2,a_2,b_2}} |g(z)|$  if  $g$  is analytic in a neighbourhood of  $E_{\rho_1,a_1,b_1,\rho_2,a_2,b_2}$ . Recall from Proposition 4.2 that  $\|f_i - Q_i\|_\infty \leq A_2[i, i]$ ,  $i = 0, \dots, N-1$ .

Once again, the diagonal right part,  $A_2$ , of the matrix will be used for controlling that the functions  $P_0(ux, v(f(x) + t))$ ,  $P_1(ux, v(f(x) + t))$ , output by the lattice basis reduction process, are uniformly small; this accounts for the presence of the  $\frac{16\rho_1\rho_2 M_{\rho_1,a_1,b_1,\rho_2,a_2,b_2}(f_i)}{(\rho_1-1)(\rho_2-1)} \left( \frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}} \right)$  remainder term.

We start with a convenient combinatorial lemma.

**Lemma 6.1.** *Let  $\gamma \geq 1$ , consider the multiset  $S = \{k + \gamma k', (k, k') \in [0, N_1 - 1] \times [0, N_2 - 1]\} + \underbrace{\{N_1, \dots, N_1\}}_{N \text{ times}}$ <sup>13</sup>, and order the elements of  $S$  as  $\sigma_0 \leq \dots \leq \sigma_{\text{card } S-1}$ .*

*Given an integer  $N$ , define*

$$\Omega_\gamma(N_1, N_2, N) = \sigma_0 + \dots + \sigma_{N-1}.$$

*Let now  $s \leq N_1$  be a real number, let  $\mathcal{K}_s = \{(i, j) \in [0, N_1 - 1] \times [0, N_2 - 1] : i + \gamma j \leq s\}$ . Then,*

$$\Omega_\gamma(N_1, N_2, N) \geq s(N - \text{card } \mathcal{K}_s) + \sum_{(i,j) \in \mathcal{K}_s} (i + \gamma j).$$

*Proof.* Put  $\mathcal{M}_s = \{i + j\gamma, (i, j) \in \mathcal{K}_s\}$ . Then, if  $\text{card } \mathcal{K}_s \leq N$ ,  $\mathcal{M}_s \subset \{\sigma_0, \dots, \sigma_{N-1}\}$ , and any element in  $\{\sigma_0, \dots, \sigma_{N-1}\} \setminus \mathcal{M}_s$  is at least equal to  $s$ .

Otherwise, we have  $\{\sigma_0, \dots, \sigma_{N-1}\} \subset \mathcal{M}_s$ , and any element in  $\mathcal{M}_s \setminus \{\sigma_0, \dots, \sigma_{N-1}\}$  is at most equal to  $s$ .  $\square$

*Remark 6.2.* The quantity  $\Omega_\gamma(N_1, N_2, N)$  plays a key role in the analysis of our bivariate method, as  $\rho_1^{\Omega_\gamma(N_1, N_2, N)}$  will turn to play the role that  $\rho^{N(N-1)/2}$  played in the univariate case. For fixed values of  $N, N_1, N_2, \gamma$ , it is easy to compute explicit values of  $\Omega_\gamma(N, N_1, N_2)$ . We thus focus in the sequel on the asymptotic (for  $N \rightarrow \infty$ ) behaviour of  $\Omega_\gamma(N, N_1, N_2)$  and shall hence mostly study the asymptotic behaviour of this bivariate method – even though the analysis itself is not asymptotic by nature (see e.g., Theorem 6.4).

We now give explicit expressions for  $\text{card } \mathcal{K}_s$  and  $\sum_{(i,j) \in \mathcal{K}_s} (i + j\gamma)$ .

**Lemma 6.3.** *Let  $s \in \mathbb{R}$ ,  $s < N_1$  and  $\gamma \geq N_1/N_2 \geq 1$ . We have*

$$(6.3) \quad \text{card } \mathcal{K}_s = \sum_{j=0}^{\lfloor s/\gamma \rfloor} (1 + \lfloor s - j\gamma \rfloor) = (1 + \lfloor s/\gamma \rfloor) + \sum_{j=0}^{\lfloor s/\gamma \rfloor} \lfloor s - j\gamma \rfloor$$

<sup>13</sup>By sum of multisets, we mean that the multiplicity of an element of the union is the sum of its multiplicities in the multisets.



and

$$(6.4) \quad \sum_{(i,j) \in \mathcal{K}_s} (i + j\gamma) = \sum_{j=0}^{\lfloor s/\gamma \rfloor} (1 + \lfloor s - j\gamma \rfloor) \left( j\gamma + \frac{\lfloor s - j\gamma \rfloor}{2} \right).$$

This implies

$$(6.5) \quad (1 + \lfloor s/\gamma \rfloor)(s - \gamma \lfloor s/\gamma \rfloor / 2) \leq \text{card } \mathcal{K}_s \leq (1 + \lfloor s/\gamma \rfloor)(1 + s - \gamma \lfloor s/\gamma \rfloor / 2)$$

and

$$(6.6) \quad (1 + \lfloor s/\gamma \rfloor) \frac{6s(s-1) + \gamma \lfloor s/\gamma \rfloor (3 - \gamma - 2\gamma \lfloor s/\gamma \rfloor)}{12} \leq \sum_{(i,j) \in \mathcal{K}_s} (i + j\gamma) \\ \leq (1 + \lfloor s/\gamma \rfloor) \frac{6s(s+1) + \gamma \lfloor s/\gamma \rfloor (3 - \gamma - 2\gamma \lfloor s/\gamma \rfloor)}{12}.$$

*Proof.* First, note that  $s/\gamma < N_1/\gamma \leq N_1/(N_1/N_2) = N_2$ , hence  $\lfloor s/\gamma \rfloor \leq N_2 - 1$ . Let  $(i, j) \in \mathcal{K}_s$ , the largest possible value of  $j$  corresponds to the case  $i = 0$ : we then have  $j\gamma \leq s$ , that is to say  $j \leq \lfloor s/\gamma \rfloor$ . Now, in order to count the elements of  $\mathcal{K}_s$ , we enumerate, for  $j = 0, \dots, \lfloor s/\gamma \rfloor$ , the elements of each slice  $\{i + j\gamma \leq s, i \in [0, N_1 - 1]\}$ : there are  $1 + \lfloor s - j\gamma \rfloor$  such elements, which proves (6.3).

Now, for  $j = 0, \dots, \lfloor s/\gamma \rfloor$ , we sum the values  $i + j\gamma$  for  $i$  in the slice  $\{i + j\gamma \leq s, i \in [0, N_1 - 1]\}$ , i.e.,  $i \in [0, \lfloor s - j\gamma \rfloor]$ . Hence, for  $j = 0, \dots, \lfloor s/\gamma \rfloor$ , we sum the values  $\lfloor s - j\gamma \rfloor(1 + \lfloor s - j\gamma \rfloor)/2 + (1 + \lfloor s - j\gamma \rfloor)j\gamma$ , from which (6.4) follows.

We use  $s - j\gamma - 1 \leq \lfloor s - j\gamma \rfloor \leq s - j\gamma$  for  $j = 0, \dots, \lfloor s/\gamma \rfloor$  to derive from (6.3)

$$(1 + \lfloor s/\gamma \rfloor)(s - \gamma \lfloor s/\gamma \rfloor / 2) \leq \text{card } \mathcal{K}_s \leq (1 + \lfloor s/\gamma \rfloor)(1 + s - \gamma \lfloor s/\gamma \rfloor / 2).$$

Likewise, we derive from (6.4)

$$\sum_{j=0}^{\lfloor s/\gamma \rfloor} \frac{(s - j\gamma)(s + j\gamma - 1)}{2} \leq \sum_{(i,j) \in \mathcal{K}_s} (i + j\gamma) \leq \sum_{j=0}^{\lfloor s/\gamma \rfloor} \frac{(1 + s - j\gamma)(s + j\gamma)}{2},$$

which yields (6.6).  $\square$

The next result gives an upper bound for the volume of the lattice generated by the rows of  $A$ :

**Theorem 6.4.** *Let  $\rho_1, \rho_2 > 1$ ,  $a_1 < b_1$ ,  $a_2 < b_2$ . We further assume that  $\rho_1^{N_1} \leq \rho_2^{N_2}$ , and define  $\gamma = \log \rho_2 / \log \rho_1$ . Let  $f_1, \dots, f_N$  be functions analytic in a neighbourhood of  $E_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}$ . Then, we have*

$$(\det A^t A)^{1/2} \leq (32\sqrt{N})^N 2^{N_1 N_2} \left( \frac{\rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \right)^N \frac{\prod_{i=1}^N M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{\rho_1^{\Omega_\gamma(N, N_1, N_2)}}.$$

*Proof.* See Appendix D.  $\square$

We now give two statements on the behaviour of  $\Omega_\gamma(N, N_1, N_2)$  when  $N \rightarrow \infty$ , for a fixed (essentially optimal) choice of  $N_1, N_2$ . The proofs are elementary, but long and we postpone them to Appendix C.

**Proposition 6.5.** *Let  $\varphi$  be the function from  $[1, +\infty)$  to  $[1, +\infty)$  defined by  $\varphi(x) = (1 + \lfloor x \rfloor)(x - \lfloor x \rfloor / 2)$ . Then  $\varphi$  is invertible. We further define  $\psi$  by*

$$\psi(x) = \frac{1 + \lfloor \varphi^{-1}(x) \rfloor}{12x} (6\varphi^{-1}(x)^2 - \lfloor \varphi^{-1}(x) \rfloor - 2\lfloor \varphi^{-1}(x) \rfloor^2);$$

we then have, for any  $y \in [1, +\infty)$ ,

$$\psi^{-1}(y) = \frac{k+1}{2} \left( 2y - k + \sqrt{4y(y-k) + 2k(2k+1)/3} \right),$$

where  $k = \lfloor 3y/2 + 1/4 \rfloor$ . Further, when  $x \rightarrow \infty$ ,  $\varphi(x) = \sqrt{2x} + O(1)$ ,  $\psi(x) = 2\sqrt{2x}/3 + O(1)$ .

*Proof.* See Appendix C.  $\square$

Note that for  $x \geq 1$ , we prove in Appendix C the inequalities  $\psi(x) - 2\sqrt{2x}/3 \in [-5/6, 0]$  and observe numerically that  $\psi(x) - 2\sqrt{2x}/3 \in [-1/2, -0.44]$ , meaning that for our purposes  $\psi(x)$  is very well approximated by  $2\sqrt{2x}/3 - 1/2$ .

**Proposition 6.6.** *Let  $\gamma \in \mathbb{R}$  such that  $3 \leq \gamma \leq N$ . Put  $s = \gamma\varphi^{-1}(N/\gamma)$ ,  $N_1 = \lfloor \sqrt{2N\gamma} \rfloor$  and  $N_2 = \lceil \sqrt{2N/\gamma} \rceil$ . Then, we have  $\gamma \geq N_1/N_2$  and*

$$\Omega_\gamma(N, N_1, N_2) = \psi(N/\gamma)N\gamma + O(N).$$

*In particular, for  $\gamma = o(N)$ , we have*

$$\Omega_\gamma(N, N_1, N_2) = \frac{2\sqrt{2}}{3} N^{3/2} \gamma^{1/2} + O(N\gamma).$$

*Proof.* See Appendix C.  $\square$

*Remark 6.7.* We can obtain a similar result for  $1 < \gamma < 3$  if we set  $N_1 = 1 + \lfloor \sqrt{2N\gamma} \rfloor$  and  $N_2 = 1 + \lceil \sqrt{2N/\gamma} \rceil$ .

This allows us to give asymptotic versions of Theorem 6.4, which will be more convenient in the sequel.

**Corollary 6.8.** *Let  $\rho_1, \rho_2 > 1$  such that  $\gamma = \log \rho_2 / \log \rho_1 \in [3, N]$ . Let  $a_1 < b_1$ ,  $a_2 < b_2$ ,  $s = \gamma\varphi^{-1}(N/\gamma)$ ,  $N_1 = \lfloor \sqrt{2N\gamma} \rfloor$  and  $N_2 = \lceil \sqrt{2N/\gamma} \rceil$ . Let  $f_1, \dots, f_N$  be functions analytic in a neighbourhood of  $E_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}$ .*

*Assume that  $N \rightarrow \infty$ , we obtain*

$$(\det A^t A)^{1/2} \leq 2^{O(N \log N)} \left( \frac{\rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \right)^N \frac{\prod_{i=1}^N M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{\rho_1^{\psi(N/\gamma)N\gamma + O(N)}}.$$

Again, we specialize this statement to the case of the functions  $f_{k,\ell}$ .

**Corollary 6.9.** *Let  $\rho_1, \rho_2 > 1$  such that  $\gamma = \log \rho_2 / \log \rho_1 \in [3, N]$ . Let  $a_1 < b_1$ ,  $a_2 < b_2$ ,  $s = \gamma\varphi^{-1}(N/\gamma)$ ,  $N_1 = \lfloor \sqrt{2N\gamma} \rfloor$  and  $N_2 = \lceil \sqrt{2N/\gamma} \rceil$ . Let  $f$  be a function analytic in a neighbourhood of  $E_{\rho, a_1, b_1}$ . Define  $f_{k,\ell}(x, t) = u^k x^k v^\ell (f(x) + t)^\ell$  for  $0 \leq \ell \leq d$ ,  $0 \leq k \leq d - \ell$ , the matrices  $A_1, A_2, A = (A_1 A_2)$  as in (6.2), and the quantity  $\Delta_{N, N_1, N_2, [a_1, b_1], [a_2, b_2], \rho_1, \rho_2} := (\det A^t A)^{1/2}$ . We have, as  $d \rightarrow +\infty$ ,*

$$(6.7) \quad \Delta_{N, N_1, N_2, [a_1, b_1], [a_2, b_2], \rho_1, \rho_2}^{1/(N-1)} \leq 2^{O(1)} \left( \sqrt{N} \frac{\rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \right)^{1+o(1)} \\ \frac{(uw)^{d/3+O(1)}}{\rho_1^{\psi(N/\gamma)\gamma + O(1)}} \left( \frac{b_1 - a_1}{2} \left( \frac{\rho_1 + \rho_1^{-1}}{2} \right) + \left| \frac{b_1 + a_1}{2} \right| \right)^{d/3+O(1)} \\ \left( M_{\rho_1, a_1, b_1}(f) + \frac{b_2 - a_2}{2} \left( \frac{\rho_2 + \rho_2^{-1}}{2} \right) + \left| \frac{b_2 + a_2}{2} \right| \right)^{d/3+O(1)}.$$

**6.2. Statement of the algorithms and proof of correctness.** Our main routine is Algorithm 4. It comes together with Algorithm 3 that mainly constructs the lattice to be reduced in Algorithm 4.

---

**Algorithm 3** Computation of the lattice to be reduced (2D approach)

---

**Input:** Four real numbers  $a_1 < b_1, a_2 < b_2$ ,  $f$  a transcendental function analytic in a complex neighbourhood of  $[a_1, b_1]$ , five nonzero integers  $d, N_1, N_2, u, v$ , two real numbers  $\rho_1, \rho_2 > 1$  such that  $16\rho_1\rho_2(\rho_1^{-N_1} + \rho_2^{-N_2}) < (\rho_1 - 1)(\rho_2 - 1)$ .

**Output:** Two matrices  $M_c \in \mathcal{M}_{N, N_1 N_2}(\mathbb{Z}), M_r \in \mathcal{M}_N(\mathbb{Z})$ , where  $N = (d+1)(d+2)/2$ , respectively storing scaled values of the coefficients and the remainders, an integer **prec** which is the precision of the computations.

```

1:  $N \leftarrow (d+1)(d+2)/2, i \leftarrow 0$ 
   // Computation of the Chebyshev nodes, listed in reverse order
2:  $L_{cheb,x} \leftarrow \left[ \frac{b_1-a_1}{2} \cos\left((j+1/2)\frac{\pi}{N_1}\right) + \frac{a_1+b_1}{2} \right]_{0 \leq j \leq N_1-1}$ 
3:  $L_{cheb,t} \leftarrow \left[ \frac{b_2-a_2}{2} \cos\left((j+1/2)\frac{\pi}{N_2}\right) + \frac{a_2+b_2}{2} \right]_{0 \leq j \leq N_2-1}$ 
4:  $M_c \leftarrow [0]_{N \times N_1 N_2}; M_r \leftarrow [0]_{N \times N}$ 
5:  $R_0 \leftarrow \frac{16(\rho_1^{1-N_1}\rho_2 + \rho_1\rho_2^{1-N_2})}{(\rho_1-1)(\rho_2-1)}, \text{prec} \leftarrow \lceil -\log_2(R_0) + \log_2(N) \rceil + 1$ 
6:  $B_x \leftarrow \left| \frac{a_1+b_1}{2} \right| + \frac{b_1-a_1}{4}(\rho_1 + \rho_1^{-1}), B_t \leftarrow \rho_2 \max(|a_2|, |b_2|)$ 
7:  $g \leftarrow (x \mapsto |f\left(\frac{a_1+b_1}{2} + \frac{b_1-a_1}{4}(\rho_1 \exp(ix) + \rho_1^{-1} \exp(-ix))\right)|)$ 
8:  $B_f \leftarrow \max(g([0, 2\pi]))$ 
9: for  $\ell = 0$  to  $d$  do
10:   for  $k = 0$  to  $d - \ell$  do
11:      $\varphi \leftarrow ((x, t) \mapsto (ux)^k (v(f(x) + t))^\ell)$ 
     // We compute the coefficient matrix : for each function, we compute its
     value at points of  $L_{cheb,x} \times L_{cheb,t}$ , use DCT and scale.
12:      $U \leftarrow \frac{4}{N_1 N_2} \text{2D-DCT-II} \left( (\varphi(L_{cheb,x}[\ell_1], L_{cheb,t}[\ell_2]))_{\substack{0 \leq \ell_1 \leq N_1-1 \\ 0 \leq \ell_2 \leq N_2-1}} \right)$ 
13:     for  $k_1 = 0$  to  $N_1 - 1$  do
14:       for  $k_2 = 0$  to  $N_2 - 1$  do
15:          $M_c[i, k_2 + k_1 N_2] \leftarrow U[k_1, k_2]$ 
16:          $M_c[i, k_2] \leftarrow \frac{1}{2} M_c[i, k_2]$ 
17:       end for
18:        $M_c[i, k_1 N_2] \leftarrow \frac{1}{2} M_c[i, k_1 N_2]$ 
19:     end for
20:     for  $j = 0$  to  $N_1 N_2 - 1$  do
21:        $M_c[i, j] \leftarrow [2^{\text{prec}} M_c[i, j]]_0$ 
22:     end for
     // We compute the scaled remainder matrix.
23:      $M_r[i, i] \leftarrow [2^{\text{prec}} R_0 (u B_x)^k (v(B_f + B_t))^\ell], i \leftarrow i + 1$ 
24:   end for
25: end for
26: Return  $M_c, M_r, \text{prec}$ 

```

---

**Algorithm 4** 2D approach to Problem 2.6

---

**Input:** Four real numbers  $a_1 < b_1, a_2 < b_2$ ,  $f$  a transcendental function analytic in a complex neighbourhood of  $[a_1, b_1]$ , five nonzero integers  $d, N_1, N_2, u, v$ , two real numbers  $\rho_1, \rho_2 > 1$  such that  $16\rho_1\rho_2(\rho_1^{-N_1} + \rho_2^{-N_2}) < (\rho_1 - 1)(\rho_2 - 1)$ .

**Output:** If successful, return a list  $\mathcal{L}$  such that  $\mathcal{L} \supset \{X \in \mathbb{Z} \text{ such that } a_1 \leq X/u \leq b_1 \text{ and there exists } Y \in \mathbb{Z}, \frac{Y}{v} \in [f(\frac{X}{u}) + a_2, f(\frac{X}{u}) + b_2]\}$ .

- 1:  $(M_c, M_r, \text{prec}) \leftarrow$  Algorithm 3  $(a_1, b_1, a_2, b_2, f, d, N_1, N_2, u, v, \rho_1, \rho_2)$ ,  $N \leftarrow (d + 1)(d + 2)/2$ ,
- 2:  $M_{LLL}, U \leftarrow$  LLL-reduce the rows of  $(M_c \mid M_r)$  // We assume that our LLL call returns both the reduced basis and the change-of-basis matrix denoted by  $U$ .
- 3: **if**  $\max(\|(M_{LLL}[0, j])_{0 \leq j \leq N+N_1N_2-1}\|_2, \|(M_{LLL}[1, j])_{0 \leq j \leq N+N_1N_2-1}\|_2) < 2^{\text{prec}}/(N + N_1N_2)$  **then**
- 4:  $L_m \leftarrow [X_1^k X_2^\ell \text{ for } k = 0 \text{ to } d - \ell \text{ for } \ell = 0 \text{ to } d]$  // List of monomials, ordered in a way compatible with Algorithm 3, Steps 9–11.
- 5:  $P_0 \leftarrow \sum_{j=0}^{N-1} U[0, j] L_m[j]$ ,  $P_1 \leftarrow \sum_{j=0}^{N-1} U[1, j] L_m[j]$ ,
- 6:  $R(X_1) \leftarrow \text{Res}_{X_2}(P_0(X_1, X_2), P_1(X_1, X_2))$
- 7: **if**  $R(X_1) \neq 0$  **then**
- 8:  $\mathcal{L} \leftarrow \{t \in \mathbb{Z}; R(t) = 0\}$
- 9: **return**  $\mathcal{L}$
- 10: **else**
- 11: **return** “FAIL”
- 12: **end if**
- 13: **else**
- 14: **return** “FAIL”
- 15: **end if**

---

Let  $[R_i, i = 0, \dots, N - 1]$  be the ordered list

$$\left[ \frac{16\rho_1\rho_2 u^k M_{\rho_1, a, b}(x)^k v^\ell M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f(x) + t)^\ell}{(\rho_1 - 1)(\rho_2 - 1)} \left( \frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}} \right); \right. \\ \left. \ell = 0, \dots, d, k = 0, \dots, d - \ell \right].$$

Let  $A = (A_1 A_2)$  and  $\hat{A} = (\hat{A}_1 \hat{A}_2)$  be the  $N \times (N_1 N_2 + N)$  matrices defined by

$$A_1 = \left( \frac{c_{k_1, k_2, i}}{2^{\delta_{0k_1} + \delta_{0k_2}}} \right)_{\substack{0 \leq i \leq N-1 \\ 0 \leq k_1 \leq N_1-1, 0 \leq k_2 \leq N_2-1}}, \quad A_2 = (\delta_{ij} R_i)_{0 \leq i, j \leq N-1}, \\ \hat{A}_1 = ([2^{\text{prec}} A_1[i, j]]_0 / 2^{\text{prec}})_{\substack{0 \leq i \leq N-1 \\ 0 \leq j \leq N_1 N_2-1}}, \quad \hat{A}_2 = ([2^{\text{prec}} A_2[i, j]] / 2^{\text{prec}})_{0 \leq i, j \leq N-1},$$

where  $\text{prec} = \lceil -\log_2(\min_{0 \leq i \leq N-1} A_2[i, i]) + \log_2(N) \rceil + 1$ . The reasons for introducing  $\hat{A}$  and  $\text{prec}$  are the same as in Section 5.

By construction,  $|\hat{A}[i, j]| \leq |A[i, j]|$  for all  $i, j$ . Hence, Theorem 6.4 and its corollaries, which proceed by upper bounding the coefficients of  $A$  and applying Theorem 5.1, also hold for  $(\det \hat{A})^{1/2}$ .

The rows of  $\hat{A}$  generate the lattice that will be reduced in our algorithm.

**Lemma 6.10.** *The  $\mathbb{Z}$ -module generated by the rows of  $\hat{A}$  is a lattice of rank  $N$ .*

*Proof.* Identical to the proof of Lemma 5.5.  $\square$

The matrices  $M_c$  and  $M_r$  computed in Algorithm 3 correspond to the scaled matrices  $2^{\text{prec}} \hat{A}_1$  and  $2^{\text{prec}} \hat{A}_2$ .

We now derive an explicit expression for  $\text{prec}$  in this bivariate context. In order to do so, we assume that the set  $u[a_1, b_1]$ , resp.  $v(f([a_1, b_1]) + [a_2, b_2])$ , contains at least one nonzero integer  $n_x$ , resp.  $n_f$ . Again, this assumption is made without loss of generality with respect to our problem, since if the assumption does not hold the problem is trivial.

**Lemma 6.11.** *We have*

$$\text{prec} = \left\lceil \log_2(1 - 1/\rho_1) + \log_2(1 - 1/\rho_2) - \log_2(\rho_1^{-N_1} + \rho_2^{-N_2}) + \log_2(N) \right\rceil - 3$$

and

$$\frac{N(\rho_1 - 1)(\rho_2 - 1)\rho_1^{N_1-1}\rho_2^{N_2-1}}{8(\rho_1^{N_1} + \rho_2^{N_2})} \leq 2^{\text{prec}} \leq \frac{N(\rho_1 - 1)(\rho_2 - 1)\rho_1^{N_1-1}\rho_2^{N_2-1}}{4(\rho_1^{N_1} + \rho_2^{N_2})}.$$

*Proof.* Under our assumption, it comes  $v(M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f(x) + t)) \geq v|n_f| \geq 1$  and  $uM_{\rho_1, a_1, b_1}(x) \geq u|n_x| \geq 1$ . It then follows  $R_i \geq 16\rho_1\rho_2(\rho_1^{-N_1} + \rho_2^{-N_2})/((\rho_1 - 1)(\rho_2 - 1)) = R_0$  for all  $i$ . Therefore, we get  $\text{prec} = \lceil -\log_2(R_{2,0}) + \log_2(N) \rceil + 1$ .  $\square$

*Remark 6.12.* All practical details and optimizations mentioned in Section 5.4 apply *mutatis mutandis* to Algorithms 3 and 4: optimization of the construction of the matrix using properties of the DCT (Section 5.4.1), overestimation issues (Section 5.4.2), rounding issues (Section 5.4.3), use of Newton polynomials (Section 5.4.4).

6.2.1. *Uniformly small polynomials in the vicinity of a transcendental analytic curve.* We now state a key result for the proof of Algorithm 4.

**Theorem 6.13.** *Let  $d \geq 1$ ,  $m \geq 2$  be two integers,  $N = (d+1)(d+2)/2$ ,  $u, v > 0$  and  $(f_j)_{1 \leq j \leq N} = (u^k v^\ell x^k (f(x) + t)^\ell)_{\substack{0 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}}$ . Let  $\rho_1, \rho_2 > 1$ ,  $a_1 < b_1$ ,  $a_2 < b_2$ ,  $N_1, N_2 \geq 1$ , and  $N \leq N_1 N_2$ . Let  $\Lambda = (\lambda_{k,\ell})_{\substack{0 \leq \ell \leq d \\ 0 \leq k \leq d-\ell}} \in \mathbb{Z}^N$  be such that  $\|\Lambda \hat{A}\|_2 < 1/(N + N_1 N_2)$ , and let  $P(X, Y) = \sum_{0 \leq k+\ell \leq d} \lambda_{k,\ell} X^k Y^\ell$ , we have*

$$\sup_{\substack{x \in [a_1, b_1] \\ t \in [a_2, b_2]}} |P(ux, v(f(x) + t))| < 1.$$

*Proof.* See Appendix D.  $\square$

We deduce the following corollary.

**Corollary 6.14.** *Under the assumptions and notations of the previous theorem, for all  $x, y$  such that  $ux, vy \in \mathbb{Z}$ , we have either  $P(ux, vy) = 0$ , or  $y \notin [f(x) + a_2, f(x) + b_2]$ .*

6.3. **Analysis.** If we apply the LLL lattice basis reduction algorithm to  $\hat{A}$ , we obtain:

**Corollary 6.15.** *Assume that  $\det(\hat{A}^t \hat{A})^{1/2(N-1)} \leq \frac{2^{-(N-1)/4 - \text{prec}/(N-1)}}{N + N_1 N_2}$ ; then Theorem 6.13 applies with  $\Lambda$  any of the first two vectors of an LLL-reduced basis of the lattice generated by the rows of  $\hat{A}$ .*

*Proof.* Identical to the proof of Corollary 5.10.  $\square$

We now study the case  $\rho_1 = K_1/(b_1 - a_1)$ , where  $K_1 > 2(b_1 - a_1)$  is a fixed real number, and similarly  $\rho_2 = K_2/(b_2 - a_2)$ ; we further assume  $\rho_1^{N_1} \leq \rho_2^{N_2}$ .

**Proposition 6.16.** *Let  $f$  be analytic in a neighbourhood of the closed disc  $\mathcal{D}_{a_1, b_1, K_1} = \{z \in \mathbb{C} : |z - (a_1 + b_1)/2| \leq K_1/2\}$ ,  $d$  be an integer  $\geq 2$ ,  $N = (d + 1)(d + 2)/2$ ,  $\rho_1 = K_1/(b_1 - a_1) > 2$ ,  $\rho_2 = K_2/(b_2 - a_2) > 2$ ,  $\gamma = \log \rho_2 / \log \rho_1 \in [3, N]$ ,  $N_1 = \lceil \sqrt{2\gamma N} \rceil$ ,  $N_2 = \lceil \sqrt{2N/\gamma} \rceil$  two integers. Let  $M_{\mathcal{D}_{a_1, b_1, K_1}}(f) := \sup_{z \in \mathcal{D}_{a_1, b_1, K_1}} |f(z)|$ .*

*Then, for  $d \rightarrow \infty$ , if*

$$b_1 - a_1 < K_1 2^{O\left(-\frac{N}{\psi(N/\gamma)\gamma}\right)} \left( \frac{uv}{2} (|a_1 + b_1| + K_1) \left( M_{\mathcal{D}_{a_1, b_1, K_1}}(f) + \frac{K_2 + |a_2 + b_2|}{2} \right) \right)^{-\frac{d}{3\psi(N/\gamma)\gamma}(1+O(1/d))},$$

*we have  $\Delta_{N, N_1, N_2, [a_1, b_1], [a_2, b_2], \rho_1, \rho_2}^{1/(N-1)} < \frac{2^{-(N-1)/4 - \text{prec}/(N-1)}}{N + N_1 N_2}$ .*

*Proof.* Note that since  $\rho_1 = K_1/(b_1 - a_1) > 2$ , we have  $E_{\rho_1, a_1, b_1} \subset \mathcal{D}_{a_1, b_1, K_1}$ . Thanks to Corollary 6.9, in view of  $(\rho_i/(\rho_i - 1))^{N/(N-1)} \leq 2^{3/2}$  for  $i \in \{1, 2\}$ , we have

$$\Delta_{N, N_1, N_2, [a_1, b_1], [a_2, b_2], \rho_1, \rho_2}^{1/(N-1)} \leq 2^{O(1)} N^{1/2+o(1)} \frac{uv(|a_1 + b_1| + K_1)/2^{d/3+O(1)}}{\rho_1^{\psi(N/\gamma)\gamma+O(1)}} \left( M_{\mathcal{D}_{a_1, b_1, K_1}}(f) + \frac{K_2 + |a_2 + b_2|}{2} \right)^{d/3+O(1)}.$$

Note that, using Lemma 6.11, as  $\rho_1^{N_1} \leq \rho_2^{N_2}$ ,  $2^{\text{prec}} \geq \frac{N}{64} \rho_1^{N_1}$ .

Thus, for  $\Delta_{N, N_1, N_2, [a_1, b_1], [a_2, b_2], \rho_1, \rho_2}^{1/(N-1)} < 2^{-(N-1)/4 - \text{prec}/(N-1)} / (N + N_1 N_2)$ , it suffices, that

$$\rho_1 > 2^{O\left(\frac{N}{\psi(N/\gamma)\gamma}\right)} \left( \frac{uv}{2} (|a_1 + b_1| + K_1) \left( M_{\mathcal{D}_{a_1, b_1, K_1}}(f) + \frac{K_2 + |a_2 + b_2|}{2} \right) \right)^{\frac{d}{3\psi(N/\gamma)\gamma}(1+O(1/d))}.$$

□

**Corollary 6.17.** *Under the assumptions of Proposition 6.16, Algorithm 4 over  $[a_1, b_1]$  and  $[a_2, b_2]$  produces at Step 5 two polynomials  $P_0, P_1$  such that*

$$\sum_{x \in [a_1, b_1], t \in [a_2, b_2]} |P_i(ux, v(f(x) + t))| < 1 \text{ for } i \in \{0, 1\}.$$

*In particular, Algorithm 4 never executes Step 14 and its output is valid.*

*Proof.* It suffices to apply Proposition 6.16, Corollary 6.15, and Theorem 6.13. □

Note again that  $P_0$  and  $P_1$  may not be coprime, in which case the algorithm returns “FAIL” at Step 17. This is what makes the algorithm heuristic.

**6.4. Complexity analysis.** In this subsection, we deduce from the bounds estimates for the complexity of our algorithm applied to a fixed interval  $[\alpha, \beta]$ . As in the univariate case, this actually requires several things:

- An evaluation of the complexity of the basic blocks, namely Algorithms 3 and 4.
- Use Corollary 6.17 to evaluate the size of a subinterval  $[a_1, b_1]$  which can be treated at once by those algorithms.

We start by giving complexity estimates for Algorithms 3 and 4.

6.4.1. *Complexity of Algorithms 3 and 4.* In this subsection, we keep notations and assumptions of Section 5.6.1.

**Proposition 6.18.** *On input  $a_1, b_1, a_2, b_2, f, d, N_1, N_2, u, v, \rho_1, \rho_2$ , under the assumption  $C_{f,p} = O(p^2)$ , the computations of Algorithm 3 can be made in floating-point precision  $p = O(\text{prec} + d \log M)$ . Hence, Steps 1–5 of Algorithm 4 have complexity  $O(d^6 M(d^2) p^2)$  using the  $L^2$  algorithm.*

*Proof.* Similar to Proposition 5.15.  $\square$

6.4.2. *Number of subintervals for fixed  $d$ .* Thanks to the results of the previous subsection, given a value  $\gamma$ , we can estimate the maximum size of an interval  $[a_1, b_1] \subset [\alpha, \beta]$ , with  $\alpha, \beta$  fixed, for which Algorithm 4 succeeds (in the sense of Corollary 6.17) and yields a lower bound of the order of magnitude  $|b - a|^{-\gamma}$ .

This follows from Proposition 6.16, and yields at the same time the number of subintervals to be considered if one wants to deal with a full interval  $[\alpha, \beta]$ .

**Theorem 6.19.** *Given fixed  $f$  and two fixed real numbers  $\alpha, \beta$ , Problem 2.6 can heuristically be solved for  $u, v \rightarrow \infty, d \rightarrow \infty, \gamma \in [3, N]$ , over  $[\alpha, \beta]$  using*

$$(6.8) \quad (\beta - \alpha) 2^{O\left(\frac{N}{\gamma \psi(N/\gamma)}\right)} (uv)^{\frac{d}{3\psi(N/\gamma)\gamma}(1+O(1/d))}$$

*calls to Algorithm 2 with parameter  $d$ .*

*We then obtain a value*

$$(6.9) \quad w = 2^{O(N/\psi(N/\gamma))} (uv)^{\frac{d}{3\psi(N/\gamma)\gamma}(1+O(1/d))}.$$

*Proof.* This is a direct consequence of Corollary 6.17, where we note that  $a_1, a_2, b_1, b_2$  are bounded, and we choose  $K_1 = 2(b_1 - a_1)$ ,  $K_2 = 1$  and  $\rho_2 = \rho_1^\gamma$ .

The heuristic nature of this result comes from the possibility that the two polynomials obtained in Algorithm 4 are not coprime, in which case one cannot recover the solutions  $X, Y$  from those two polynomials.

Finally, the lower bound  $b_2 - a_2$  is  $O((b_1 - a_1)^\gamma)$ , from which the second part of the result follows.  $\square$

*Remark 6.20.* To get a better feeling of this result we should distinguish two cases:

- We let  $\gamma$  tend to infinity as  $N/\kappa, \kappa \geq 1$ . Then, we obtain an upper bound for the number of intervals  $O\left((\beta - \alpha)(uv)^{-\frac{2\kappa}{3d\psi(\kappa)}(1+O(1/d))}\right)$ , with  $w = (uv)^{\frac{d}{3\psi(\kappa)}(1+O(1/d))}$ .
- If, on the other hand,  $\gamma = o(N)$ ,  $N/\gamma$  tends to infinity and we can use the asymptotic estimate  $\psi(N/\gamma) = \frac{2}{3}\sqrt{\frac{2N}{\gamma}} + O(1)$  to get a bound on the number  $n_I$  of intervals and on  $w$  of the respective forms

$$\begin{aligned} n_I &= (\beta - \alpha) 2^{O(d/\sqrt{\gamma})} (uv)^{1/(2\sqrt{\gamma})(1+O(\sqrt{\gamma}/d))}, \\ w &= 2^{O(d)} (uv)^{\sqrt{\gamma}/2(1+O(\sqrt{\gamma}/d))}. \end{aligned}$$

The first case resembles the results obtained in the previous section with the univariate algorithm (but with a different constant), whereas the second part is unattainable using the methods of the previous section.

For  $u = v = 2^p, \gamma = 4 + o(1), d = o(p)$ , we recover Stehlé's result [59], namely the fact that we can solve the TMD (i.e., get the bound  $1/w = 2^{-p}$  in time  $2^{p/2(1+o(1))}$ ).

*Remark 6.21.* One can adapt Remark 5.20, Theorems 5.23 and 5.24 in the case of this bivariate method. However, for the last two results, the region where the discussion makes sense is restricted to  $\gamma \gg N/(\log N)^2$ , as otherwise the impact of choosing an optimal  $\rho_1$  occurs only on second order terms.

Further, this leads to a somewhat delicate discussion and, in the end, yields the same result up to some improvement in the constants. We thus chose not to include the corresponding theorems.

## 7. COMPARISON WITH PREVIOUS WORK

The following table summarizes the main results of the paper and compares them to [59]. The columns complexity and bound on  $w$  should be understood as the exponent of  $uv$  in the corresponding values. For the sake of readability,

- In the first row, we restrict to  $\omega_0 \in \mathbb{Z}_{>0}$  just in order to get a more compact bound ;
- In the fourth row, we shall assume that  $d = o(\log(uv))$  ;
- We shall omit all factors  $(1 + o(1))$  in the asymptotic results.

Rows labelled “1D” refer to Section 5 (Algorithms 1 and 2) whereas rows labelled “2D” refer to Section 6 (Algorithms 3 and 4).

	References	Parameters to be chosen	Matrix dimensions	Complexity	Bound on $w$
1D	Rk. 5.18	$\omega_0 \in [0, N] \cap \mathbb{Z}_{>0}$	$N \times 2N$	$\frac{2Nd}{(N-\omega_0)(N+\omega_0-3)}$	$\frac{2Nd}{N+\omega_0-3}$
1D $_{d \rightarrow \infty}$	Cor. 5.19	$\lambda = \omega_0/N \in [0, 1]$	$N \times 2N$	$\frac{4}{3(1-\lambda^2)d}$	$\frac{2d}{1+\lambda}$
2D $_{d \rightarrow \infty}$	Rk. 6.20	$\gamma = N/\kappa, \kappa \geq 1$	$\approx N \times 3N$	$\frac{2\kappa}{3d\psi(\kappa)}$	$\frac{d}{3\psi(\kappa)}$
2D $_{d \rightarrow \infty}$	Rk. 6.20	$\gamma = o(N)$	$\approx N \times 3N$	$\frac{1}{2\sqrt{\gamma}}$	$\sqrt{\gamma}/2$
S $_{\alpha \rightarrow \infty}$	[59]	$\xi$	$\frac{(\alpha+1)(\alpha+2)}{2} \times O(\xi\alpha^2)$	$\frac{1}{2\sqrt{\xi}}$	$\sqrt{\xi}/2$

TABLE 2. Comparison of the main methods of this paper and [59]

Concerning [59], we have introduced a parameter  $\xi$  for a clearer comparison; namely, we have put (with Stehlé’s notations)  $m - t = 1/2\sqrt{\xi}$ , which gives  $n_2 + t = \sqrt{\xi}/2$ . Note that Stehlé’s  $\alpha$  plays a role similar to our  $d$ .

*Remark 7.1.* Table 2 only estimates the exponential part of the complexity; it would remain to estimate the polynomial part, which is dominated by the cost of lattice basis reduction. Akhavi-Stehlé’s trick reduces greatly the influence of the dimension, but the size of the integers involved in the different methods differ. Roughly speaking,

- in the case of the univariate method, the size of the integers involved is  $\approx \text{prec} \approx N \log \rho$ , which is of the order of the bound on  $w$ ;
- in the case of the bivariate method, the size of the integers involved is  $\approx \text{prec} \approx \log \max(\rho_1^{N_1}, \rho_2^{N_2})$ , whereas the bound on  $w$  is of the order of  $\rho_2$ ; as we expect, for optimal choices of parameters, that  $N_1 \log \rho_1$  and  $N_2 \log \rho_2$  have the same order of magnitude, this size is thus  $\approx N_2 \log \rho_2 \approx N_2 \log w$ . As  $N_2 \approx d/\sqrt{\gamma}$ , this is  $d \log w/\sqrt{\gamma}$ .



- in Stehlé’s paper, the integers involved are of the order of  $M^\alpha$ , which, in our notations, means that their size is of the order of  $d \log w$ .

This difference in the size of the integers involved may, at least partially, explain the fact that lattice basis reduction performs somewhat better in our method than in the BaCSeL implementation of Stehlé’s method (see Section 8).

**7.1. Univariate method and Bombieri and Pila’s approach.** Our univariate method bears a strong resemblance to Bombieri & Pila’s approach [6] of bounding the number of integer points on an analytic curve. The method that we develop is effective, and yields a way to not only control integer points on the curve, but close to the curve.

Note that we recover Bombieri and Pila’s estimate [6, Main Lemma] under the form  $(uv)^{\frac{4}{3(d+3)}}$ , which is the number of intervals required. We can thus recover, following Bombieri and Pila’s arguments based on [6, Thm. 5] or [54], their bound for the number of points on the curve (without any heuristic). Our method also allows for the explicit determination of those points, but is on this point only (mildly) heuristic.

Our variation with the  $\omega_0$ , on the other hand, is new; it worsens the quality of Bombieri-Pila’s bound, but improves the distance around the curve in which we are able to detect points with denominator dividing  $v$ .

**7.2. Bivariate method vs. Stehlé’s approach.** On the other hand, our bivariate method bears a strong resemblance with Stehlé’s work [59]. Our approach mostly differs by the use of approximation-related techniques (Chebyshev interpolants) rather than a computer-algebra oriented vision of functions using Taylor expansions.

We have a dense representation of our auxiliary polynomials, which leads us to manipulate almost square matrices, whereas Stehlé’s matrices are inherently more rectangular, because he has to represent all coefficients of the bivariate polynomials he manipulates.

Table 5 shows that our approach is better in practice. We propose an analysis of this favourable situation in Section 8.3.

Note two further facts :

- our analysis is sharper in the case of entire functions, as we are able to take into account the growth of the function at infinity. The same results could probably be derived in Stehlé’s paper using Cauchy inequalities to estimate Taylor coefficients.

This sharper analysis allows us to obtain, in Problem 2.6, a lower bound  $1/w \geq (uv)^{-O(p^2/\log p)}$  in the case where we want to solve the problem without any subdivision. This improves at the same time on Stehlé’s method and Nesterenko-Waldschmidt paper which, though with different goals and through a purely theoretical (vs. algorithmic) method, both obtain the lower bound  $w \leq (uv)^{-O(p^2)}$ .

- Our analysis is also sharper in the more practical domain where we choose  $\gamma = N/\kappa$  for some constant  $\kappa$ . We obtain better constants in the exponents for both the overall complexity of the method and the bound on  $w$ .

## 8. EXPERIMENTAL RESULTS

We have implemented TMD-oriented versions of our algorithms in SageMath<sup>14</sup>. Our codes are available from <https://perso.ens-lyon.fr/nicolas.brisebarre/tmd.html>. The tests hereafter were executed on an Intel Xeon E5620 2.40GHz CPU with a 64-bit Linux-based system.

In the two examples that we address, we cut the binades under consideration into subintervals of the same size and we apply the algorithms to each subinterval. For Algorithms 1 and 2, the subinterval will correspond to the interval  $[a, b]$  considered in these algorithms. For Algorithms 3 and 4, the subinterval will correspond to the interval  $[a_1, b_1]$ , while  $[a_2, b_2]$  will be equal to  $[-1/w, 1/w]$ , cf. Problem 2.6.

Currently, the main expensive part of our algorithms is the LLL reduction. In our implementations, the following two optimizations led to a significant speedup of the LLL reduction part:

- (1) we use a random projection trick inspired from [2], cf. Theorem 4.5.
- (2) If we consider two contiguous subintervals  $I_0$  and  $I_1$ , the matrices  $M_{c,I_0}$  and  $M_{r,I_0}$ ,  $M_{c,I_1}$  and  $M_{r,I_1}$ , output by Algorithm 1 (resp. Algorithm 3) applied to  $I_0$  and  $I_1$  will be close by construction. Hence, our optimization consists in:
  - retrieving the change-of-basis matrix  $U_{I_0}$  computed at Step 2 of Algorithm 2 (resp. Step 2 of Algorithm 4) applied to  $I_0$ ,
  - then left-multiplying  $M_{c,I_1}$  and  $M_{r,I_1}$  by  $U_{I_0}$ , which operates in practice as a significant prereduction of the lattice built from  $M_{c,I_1}$  and  $M_{r,I_1}$ ,
  - eventually, we apply LLL to these prerduced matrices.

Another optimization comes from the use of Newton polynomials instead of monomials (as pointed in Section 5.4.4): for given values of  $d, N, N_1, N_2$ , it makes it possible to process larger subintervals.

The timings presented with a \* are estimated ones: we performed our computations on a subinterval and then extrapolate the timing to address the whole binade.

We chose to limit the evaluation of our algorithms on feasible computations in `binary128`, namely computations that could be performed in real life, possibly using a large cluster. In terms of the bound on  $w$ , we have thus excluded the optimal case of the TMD, namely  $w \approx 2^{2p}$ , and have started at  $w \approx 2^{6p}$ .

**8.1. Algorithms 1 and 2 in action: the TMD for the gamma function in `binary128`.** Euler's gamma [63, Chap. 3] is one of the functions of the C mathematical library. Very little is known about the Diophantine properties of its values at rational numbers: we have  $\Gamma(k+1) = k!$  for any  $k \in \mathbb{N}$  and the transcendence of the numbers  $\Gamma(1/2), \Gamma(1/3), \Gamma(1/4), \Gamma(1/6), \Gamma(2/3), \Gamma(3/4), \Gamma(5/6)$  [68]. We used our implementation of Algorithms 1 and 2 to address the TMD over  $[1/2, 1)$ , for directed rounding functions and for the precision  $p = 113$ . More precisely, we address the following question, for various values of the parameters  $d$  and  $\omega_0$ : compute  $w > 0$  and all the integers  $X$ ,  $1 \leq X/2^{p-1} < 2$  for which there exists  $Y \in \mathbb{Z}$  satisfying

$$\left| \Gamma\left(\frac{X}{2^{p-1}}\right) - \frac{Y}{2^p} \right| < \frac{1}{w}.$$

<sup>14</sup><https://www.sagemath.org/>

We report in Table 3 our results. We first set  $p = 113$ ,  $u = 2^{p-1}$ ,  $v = 2^p$ , then the integer  $d$  and the real number  $\omega_0$ , and finally we choose  $\rho$  in order to maximize the width of the subinterval  $[a, b]$  of  $[1, 2)$  which we apply the algorithms to.

$d$	$N$	$\rho$	$b - a$	$\omega_0$	$\log_2(w)$	Timing
6	28	$2^{32}$	$2^{-31}$	0	$7.86p^*$	35.53* years
8	45	$2^{26}$	$21/2^{29}$	0	$10.11p^*$	510* days
10	66	$2^{21}$	$13/2^{24}$	0	$12.24p$	102 days
12	91	$2^{20}$	$13/2^{22}$	20	$12.78p$	82* days
12	91	$2^{20}$	$17/2^{22}$	14	$13.32p$	76* days
12	91	$2^{18}$	$15/2^{21}$	0	$14.44p$	46 days

TABLE 3. The gamma function over the binade  $[1, 2)$

**8.2. Using  $\omega_0$  in Algorithms 1 and 2 : the TMD for the exponential function in binary128 - experimental validation of Figure 1.** Using the exp function over  $[1/4, 1/2)$ , we have also studied the influence of  $\omega_0$ , in order to build the experimental equivalent of the theoretical curves of Figure 1.

The exponential function is also part of the C mathematical library. We recall that Section 3.1 presents up-to-date, to the best of our knowledge, theoretical results regarding the TMD in the case of the exponential function.

Here, we address the TMD for exp over  $[1/4, 1/2)$ , for directed rounding functions and for the precision  $p = 113$ : we compute  $w > 0$  and all the integers  $X$ ,  $1/4 \leq X/2^{p+1} < 1/2$  for which there exists  $Y \in \mathbb{Z}$  satisfying

$$\left| \exp\left(\frac{X}{2^{p+1}}\right) - \frac{Y}{2^{p-1}} \right| < \frac{1}{w}.$$

Here, we set  $u = 2^{p+1}$  and  $v = 2^{p-1}$ .

For each value of the bound on  $w$  we have tried to find, for various values of  $d$ , the  $\omega_0$  allowing to attain this bound in minimal time (in the case of exp, we use  $\rho = d/(b - a)$ , cf. Theorem 5.23). The following picture represents the  $\log_2$  of the time to treat a binade ( $y$ -axis) as a function of  $\log_2(w)/p$  ( $x$ -axis); this is indeed the experimental equivalent of Figure 1, up to a rescaling of the  $x$ -axis (by a factor of 2, as Figure 1 is expressed in terms of powers of  $uv = 2^{2p}$ ) and  $y$ -axis.

**8.3. Algorithms 3 and 4 in action: the TMD for the exponential function in binary128.** We used our implementation of Algorithms 3 and 4 to address the TMD over  $[1/4, 1/2)$ , for directed rounding functions and for the precision  $p = 113$ . More precisely, we address the following question, for various values of the parameter  $w$ : determine the integers  $X$ ,  $1/4 \leq X/2^{p+1} < 1/2$  for which there exists  $Y \in \mathbb{Z}$  satisfying

$$\left| \exp\left(\frac{X}{2^{p+1}}\right) - \frac{Y}{2^{p-1}} \right| < \frac{1}{w}.$$

We report in Table 4 our results. We first set  $p = 113$ ,  $u = 2^{p+1}$ ,  $v = 2^{p-1}$ ,  $b_2 = -a_2 = 1/w$  and the value of  $d$ . The choice of the parameters  $N_1, N_2, \rho_1, a_1$  and  $b_1$ <sup>15</sup> is then made in order to maximize the width of the subinterval  $[a_1, b_1]$  of  $[1/4, 1/2)$ . We finally fix  $\rho_2 = \min(\rho_1^{N_1/N_2}, 1/b_2)$ .

<sup>15</sup>Actually, it is the value of  $b_1 - a_1$  which matters and not the values of  $a_1$  and  $b_1$ .

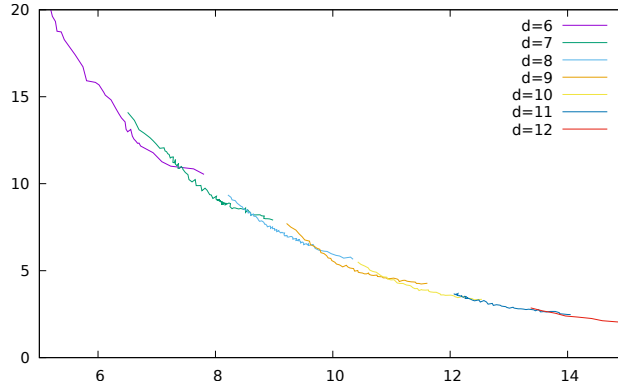


FIGURE 2. Exponent estimates

$\log_2(w)$	$d$	$N$	$N_1$	$N_2$	$\rho_1$	$b_1 - a_1$	Timing
$6p$	6	28	24	2	$2^{36}$	$7/2^{35}$	22.63* years
$6p$	12	91	60	3	$2^{29}$	$59/2^{30}$	3.41* years
$8p$	8	45	39	2	$2^{30}$	$15/2^{29}$	203 days
$8p$	12	91	59	3	$2^{26}$	$2^{-21}$	137 days
$10p$	10	66	59	2	$2^{25}$	$7/2^{23}$	25.8 days
$10p$	12	91	61	3	$2^{24}$	$7/2^{22}$	37.7 days
$12p$	12	91	80	2	$2^{21}$	$5/2^{19}$	8.7 days

TABLE 4. The exponential function over the binade  $[1/4, 1/2)$ 

For the sake of practical comparison, we have attempted a comparison with BaCSeL-4.0<sup>16</sup>, which implements [59]. For the latter, we used BaCSeL-4.0 only for the generation of the corresponding matrix, and simply measured the cost of the LLL step (which dominates the total cost anyway), using the same implementation of fpLLL as in our code (BaCSeL is bundled with an ad hoc lattice basis reduction implementation, corresponding to an old fpLLL version).

For the comparison to be fair, we have included the Akhavi-Stehlé’s trick (cf. Theorem 4.5) in BaCSeL. We have also tried to include the prereluction trick but the latter, in the setting of [59] seems to make the reduction more costly. In this case, the complete timings are merely estimates for the cost of treating a whole binade.

The results are reported in Table 5. We recall that BaCSeL’s  $d$  is a technical parameter with a completely different meaning of our  $d$ , which corresponds to BaCSeL’s  $\alpha$ ; finally, BaCSeL’s  $t$  corresponds to our notations  $p - 1 - \log_2(b_1 - a_1)$ . To avoid any confusion, we will denote them  $\alpha_{\text{Ste}}$ ,  $d_{\text{Ste}}$  and  $t_{\text{Ste}}$ .

We observe that we gain a significant constant factor, increasing with the value of  $d$ ; our method allows for slightly larger intervals (by a factor around 2, which seems to decrease with the degree), but the main factors explaining the difference are the fact that our lattices seem somewhat easier to reduce and that the “prereluction

<sup>16</sup><https://gitlab.inria.fr/zimmerma/bacsel>

$\log_2(w)$	$\alpha_{\text{Ste}}$	$d_{\text{Ste}}$	$t_{\text{Ste}}$	Timing	Comparison with this paper
$6p$	6	16	78.7	169* years	$\times 7.5$
$6p$	12	20	86.2	334* years	$\times 98$
$8p$	8	30	86.3	16.82* years	$\times 30$
$8p$	12	30	90.1	35.01* years	$\times 93$
$10p$	10	40	90.8	7.37* years	$\times 104$
$10p$	12	55	93.2	10.53* years	$\times 102$
$12p$	12	60	94.6	3.81* years	$\times 160$

TABLE 5. Stehlé’s BaCSeL parameters and timings for the exponential function over the binade  $[1/4, 1/2)$

trick” also plays an important role. These three terms each account for a small 2 to 5 (depending on the cases) factor, explaining overall the factors  $\approx 9$ -100 that we observe above.

It should finally be recalled that the comparison is biased in favour of [59] : we are comparing optimized C code to an algorithm implemented in an interpreted language. The comparison remains rather fair as long as the lattice basis reduction dominates (which is the case for Stehlé’s  $\alpha_{\text{Ste}} = 10, 12$ ) but a low-level implementation of our algorithms is required in order to get reliable results when our parameter  $d \leq 8$ , where the gap is probably larger than suggested by Table 4.

*Remark 8.1.* Note that a more relevant comparison between Table 4 and Table 5 would probably be obtained by comparing

- Row 2 of Table 4 with Row 1 of Table 5, which corresponds to the best choice of parameters for the problem with  $\log_2(w) = 6p$  ; with this criterion, the ratio is  $\approx 50$ .
- Row 4 of Table 4 with Row 3 of Table 5, which corresponds to the best choice of parameters for the problem with  $\log_2(w) = 8p$  ; with this criterion, the ratio is  $\approx 45$ .

## 9. CONCLUSION

We expect this work to be used in practice to address the TMD for the binary128 format, but we also hope that it will be of practical use for the determination of integer points close to a transcendental curve. Moreover, we believe that the tools that we have developed are of a more general interest in the context of practical applications of Coppersmith’s method, and allow easier analysis of some “rectangular” variants.

Regarding future work, we plan, first, to improve our current implementations and then to study the case of algebraic functions, for which we have some preliminary results.

## ACKNOWLEDGEMENTS.

We wish to thank Jean-Michel Muller for so many invaluable discussions and Martin Albrecht for his kind and effective help about fpLLL.

## REFERENCES

- [1] M. Ajtai. The Shortest Vector Problem in  $L_2$  is NP-hard for Randomized Reductions (Extended Abstract). In *Proceedings of the 30th ACM symposium on Theory of computing (STOC)*, pages 10–19, 1998.
- [2] A. Akhavi and D. Stehlé. Speeding-up Lattice Reduction with Random Projections. In E. S. Laber, C. F. Bornstein, L. T. Nogueira, and L. Faria, editors, *LATIN 2008: Theoretical Informatics, 8th Latin American Symposium, Búzios, Brazil, April 7-11, 2008, Proceedings*, volume 4957 of *Lecture Notes in Computer Science*, pages 293–305. Springer, 2008.
- [3] American National Standards Institute and Institute of Electrical and Electronic Engineers. *IEEE Standard for Binary Floating-Point Arithmetic*. ANSI/IEEE Standard 754–1985, 1985.
- [4] American National Standards Institute and Institute of Electrical and Electronic Engineers. *IEEE Standard for Radix Independent Floating-Point Arithmetic*. ANSI/IEEE Standard 854–1987, 1987.
- [5] V. Berthé and L. Imbert. Diophantine approximation, Ostrowski numeration and the double-base number system. *Discrete Math. Theor. Comput. Sci.*, 11(1):153–172, 2009.
- [6] E. Bombieri and J. Pila. The number of integral points on arcs and ovals. *Duke Math. J.*, 59(2):337–357, 1989.
- [7] D. Boneh and G. Durfee. Cryptanalysis of RSA with private key  $d$  less than  $N^{0.292}$ . *IEEE Trans. Inform. Theory*, 46(4):1339–1349, 2000.
- [8] J. P. Boyd. *Chebyshev and Fourier spectral methods*. Dover Publications Inc., Mineola, NY, second edition, 2001. Available from [http://www-personal.umich.edu/~jpbayd/B00K\\_Spectral2000.html](http://www-personal.umich.edu/~jpbayd/B00K_Spectral2000.html).
- [9] N. Brisebarre and S. Chevillard. Efficient polynomial  $L^\infty$  approximations. In *ARITH '07: Proceedings of the 18th IEEE Symposium on Computer Arithmetic*, pages 169–176, Washington, DC, 2007. IEEE Computer Society.
- [10] N. Brisebarre, G. Hanrot, and O. Robert. Exponential sums and correctly-rounded functions. *IEEE Trans. Comput.*, 66(12):2044–2057, 2017.
- [11] N. Brisebarre and J.-M. Muller. Correct rounding of algebraic functions. *Theor. Inform. Appl.*, 41(1):71–83, 2007.
- [12] M. Bronstein. *Symbolic integration. I*, volume 1 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, second edition, 2005. Transcendental functions, With a foreword by B. F. Caviness.
- [13] J. W. S. Cassels. *An introduction to the geometry of numbers*. Classics in Mathematics. Springer-Verlag, Berlin, 1997. Corrected reprint of the 1971 edition.
- [14] E. W. Cheney. *Introduction to approximation theory*. AMS Chelsea Publishing, Providence, RI, 1998. Reprint of the second (1982) edition.
- [15] S. Chevillard. *Évaluation efficace de fonctions numériques – Outils et exemples*. PhD thesis, École normale supérieure de Lyon – Université de Lyon, 2009. In French.
- [16] W. J. Cody. A proposed radix and word length independent standard for floating-point arithmetic. *ACM SIGNUM Newsletter*, 20:37–51, Jan. 1985.
- [17] W. J. Cody, J. T. Coonen, D. M. Gay, K. Hanson, D. Hough, W. Kahan, R. Karpinski, J. Palmer, F. N. Ris, and D. Stevenson. A proposed radix-and-word-length-independent standard for floating-point arithmetic. *IEEE MICRO*, 4(4):86–100, Aug. 1984.
- [18] H. Cohen. *A course in computational algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 1993.
- [19] D. Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *J. Cryptology*, 10(4):233–260, 1997.
- [20] D. Coppersmith. Finding small solutions to small degree polynomials. In J. H. Silverman, editor, *Proceedings of Cryptography and Lattices (CaLC)*, volume 2146 of *Lecture Notes in Computer Science*, pages 20–31. Springer-Verlag, Berlin, 2001.
- [21] F. de Dinechin, A. V. Ershov, and N. Gast. Towards the post-ultimate libm. In *Proceedings of the 17th IEEE Symposium on Computer Arithmetic*, ARITH '05, pages 288–295, Washington, DC, USA, 2005. IEEE Computer Society.
- [22] F. de Dinechin, C. Lauter, and J.-M. Muller. Fast and correctly rounded logarithms in double-precision. *Theor. Inform. Appl.*, 41(1):85–102, 2007.
- [23] F. De Dinechin, J.-M. Muller, B. Pasca, and A. Plesco. An FPGA architecture for solving the Table Maker's Dilemma. In *Application-Specific Systems, Architectures and Processors*

- (ASAP), 2011 IEEE International Conference on, pages 187–194, Santa Monica, United States, Sept. 2011. IEEE Computer Society.
- [24] L. Fox and I. B. Parker. *Chebyshev polynomials in numerical analysis*. Oxford University Press, London-New York-Toronto, Ont., 1968.
- [25] F. Gramain. Sur le lemme de Siegel (d’après E. Bombieri et J. Vaaler). *Publications mathématiques de l’Univ. P. et M. Curie*, 64:fascicule 1, 1983-1984.
- [26] P. M. Gruber and C. G. Lekkerkerker. *Geometry of numbers*, volume 37 of *North-Holland Mathematical Library*. North-Holland Publishing Co., Amsterdam, second edition, 1987.
- [27] IEEE. *IEEE Standard for Floating-Point Arithmetic (IEEE Std 754-2019)*. July 2019. Available at <https://ieeexplore.ieee.org/servlet/opac?punumber=8766227>.
- [28] IEEE Computer Society. *IEEE Standard for Floating-Point Arithmetic*. IEEE Standard 754-2008, Aug. 2008. Available at <https://ieeexplore.ieee.org/servlet/opac?punumber=4610933>.
- [29] C. Iordache and D. W. Matula. On infinitely precise rounding for division, square root, reciprocal and square root reciprocal. In Koren and Kornerup, editors, *Proceedings of the 14th IEEE Symposium on Computer Arithmetic (Adelaide, Australia)*, pages 233–240. IEEE Computer Society Press, Los Alamitos, CA, Apr. 1999.
- [30] C.-P. Jeannerod, N. Louvet, J.-M. Muller, and A. Panhaleux. Midpoints and exact points of some algebraic functions in floating-point arithmetic. *IEEE Trans. Comput.*, 60(2):228–241, 2011.
- [31] F. Johansson. Arb: efficient arbitrary-precision midpoint-radius interval arithmetic. *IEEE Trans. Comput.*, 66(8):1281–1292, 2017.
- [32] W. Kahan. Why do we need a floating-point standard? Technical report, Computer Science, UC Berkeley, 1981. Available at <https://www.cs.berkeley.edu/~wkahan/ieee754status/why-ieee.pdf>.
- [33] S. Khémira and P. Voutier. Approximation diophantienne et approximations de Hermite-Padé de type I de fonctions exponentielles. *Ann. Sci. Math. Québec*, 35(1):85–116, 2011.
- [34] S. Khémira. *Approximants de Hermite-Padé, déterminants d’interpolation et approximation diophantienne*. PhD thesis, Université Paris 6, Paris, France, 2005.
- [35] O. Knill. Cauchy-Binet for pseudo-determinants. *Linear Algebra Appl.*, 459:522–547, 2014.
- [36] T. Lang and J.-M. Muller. Bound on run of zeros and ones for algebraic functions. In N. Burgess and L. Ciminiera, editors, *Proceedings of the 15th IEEE Symposium on Computer Arithmetic (ARITH-16)*, pages 13–20, June 2001.
- [37] C. Q. Lauter. *Arrondi Correct de Fonctions Mathématiques*. PhD thesis, École Normale Supérieure de Lyon, Lyon, France, Oct. 2008.
- [38] V. Lefèvre. *Moyens Arithmétiques Pour un Calcul Fiable*. PhD thesis, École Normale Supérieure de Lyon, Lyon, France, 2000.
- [39] V. Lefèvre. New results on the distance between a segment and  $\mathbb{Z}^2$ . Application to the exact rounding. In *Proceedings of the 17th IEEE Symposium on Computer Arithmetic (ARITH-17)*, pages 68–75. IEEE Computer Society Press, Los Alamitos, CA, June 2005.
- [40] V. Lefèvre and J.-M. Muller. Worst cases for correct rounding of the elementary functions in double precision. In N. Burgess and L. Ciminiera, editors, *Proceedings of the 15th IEEE Symposium on Computer Arithmetic (ARITH-16)*, pages 111–118, Vail, CO, June 2001.
- [41] V. Lefèvre, J.-M. Muller, and A. Tisserand. Towards correctly rounded transcendentals. In *Proceedings of the 13th IEEE Symposium on Computer Arithmetic*, pages 132–137. IEEE Computer Society Press, Los Alamitos, CA, 1997.
- [42] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261(4):515–534, 1982.
- [43] J. Liouville. Nouvelle démonstration d’un théorème sur les irrationnelles algébriques. *C. R. Acad. Sci. Paris*, 18:910–911, 1844.
- [44] J. Liouville. Remarques relatives à des classes très-étendues de quantités dont la valeur n’est ni algébrique ni même réductible à des irrationnelles algébriques. *C. R. Acad. Sci. Paris*, 18:883–885, 1844.
- [45] J. Liouville. Sur des classes très étendues de quantités dont la valeur n’est ni algébrique ni même réductible à des irrationnelles algébriques. *J. Math. Pures Appl.*, 16:133–142, 1851.
- [46] L. Lovász. *An algorithmic theory of numbers, graphs and convexity*, volume 50 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1986.

- [47] P. W. Markstein. The New IEEE-754 Standard for Floating-Point Arithmetic. In *Numerical Validation in Current Hardware Architectures*, number 08021 in Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2008.
- [48] J. C. Mason and D. C. Handscomb. *Chebyshev polynomials*. CRC Press, 2002.
- [49] J.-M. Muller. *Elementary Functions, Algorithms and Implementation*. Birkhäuser, Boston, 3rd edition, 2016.
- [50] J.-M. Muller, N. Brisebarre, F. de Dinechin, C.-P. Jeannerod, V. Lefèvre, G. Melquiond, N. Revol, D. Stehlé, and S. Torres. *Handbook of Floating-Point Arithmetic*. Birkhäuser, 2010.
- [51] Y. V. Nesterenko and M. Waldschmidt. On the approximation of the values of exponential function and logarithm by algebraic numbers (in Russian). *Mat. Zapiski*, 2:23–42, 1996. Available in English at <http://www.math.jussieu.fr/~miw/articles/ps/Nesterenko.ps>.
- [52] P. Q. Nguyen and D. Stehlé. An LLL algorithm with quadratic complexity. *SIAM J. Comput.*, 39(3):874–903, 2009.
- [53] P. Q. Nguyen and B. Vallée, editors. *The LLL Algorithm - Survey and Applications*. Information Security and Cryptography. Springer, 2010.
- [54] J. Pila. Density of integer points on plane algebraic curves. *Internat. Math. Res. Notices*, (18):903–912, 1996.
- [55] G. Plonka, D. Potts, G. Steidl, and M. Tasche. *Numerical Fourier analysis*. Applied and Numerical Harmonic Analysis. Birkhäuser/Springer, Cham, 2018.
- [56] M. J. D. Powell. *Approximation theory and methods*. Cambridge University Press, 1981.
- [57] T. J. Rivlin. *The Chebyshev polynomials*. Wiley-Interscience [John Wiley & Sons], New York-London-Sydney, 1974. Pure and Applied Mathematics.
- [58] M. J. Schulte and E. E. Swartzlander. Exact rounding of certain elementary functions. In E. E. Swartzlander, M. J. Irwin, and G. Jullien, editors, *Proceedings of the 11th IEEE Symposium on Computer Arithmetic*, pages 138–145. IEEE Computer Society Press, Los Alamitos, CA, June 1993.
- [59] D. Stehlé. On the randomness of bits generated by sufficiently smooth functions. In F. Hess, S. Pauli, and M. E. Pohst, editors, *Proceedings of the 7th Algorithmic Number Theory Symposium, ANTS VII*, volume 4076 of *Lecture Notes in Computer Science*, pages 257–274. Springer-Verlag, Berlin, 2006.
- [60] D. Stehlé, V. Lefèvre, and P. Zimmermann. Worst Cases and Lattice Reduction. In J.-C. Bajard and M. J. Schulte, editors, *Proceedings of the 16th IEEE Symposium on Computer Arithmetic (ARITH-16)*, pages 142–147. IEEE Computer Society Press, Los Alamitos, CA, June 2003.
- [61] D. Stehlé, V. Lefèvre, and P. Zimmermann. Searching Worst Cases of a One-Variable Function Using Lattice Reduction. *IEEE Trans. Comput.*, 54(3):340–346, Mar. 2005.
- [62] G. Strang. The discrete cosine transform. *SIAM Rev.*, 41(1):135–147, 1999.
- [63] N. M. Temme. *Special functions*. A Wiley-Interscience Publication. John Wiley & Sons, Inc., New York, 1996. An introduction to the classical functions of mathematical physics.
- [64] S. Torres. *Tools for the Design of Reliable and Efficient Functions Evaluation Libraries*. PhD thesis, École normale supérieure de Lyon – Université de Lyon, Lyon, France, 2016.
- [65] L. N. Trefethen. *Approximation Theory and Approximation Practice*. SIAM, 2013.
- [66] W. Tucker. *Validated numerics, A short introduction to rigorous computations*. Princeton University Press, Princeton, NJ, 2011. A short introduction to rigorous computations.
- [67] J. von zur Gathen and J. Gerhard. *Modern computer algebra*. Cambridge University Press, third edition, 2013.
- [68] M. Waldschmidt. Transcendence of periods: the state of the art. *Pure Appl. Math. Q.*, 2(2, Special Issue: In honor of John H. Coates. Part 2):435–463, 2006.
- [69] K. Xu. The Chebyshev points of the first kind. *Appl. Numer. Math.*, 102:17–30, 2016.
- [70] A. Ziv. Fast evaluation of elementary mathematical functions with correctly rounded last bit. *ACM Trans. Math. Software*, 17(3):410–423, Sept. 1991.
- [71] A. Zygmund. *Trigonometric series. Vol. I, II*. Cambridge Mathematical Library. Cambridge University Press, Cambridge, third edition, 2002.



## APPENDIX A. PROOFS OF FACTS REGARDING CHEBYSHEV POLYNOMIALS

First, we recall the aliasing phenomenon in the case of Chebyshev nodes of the first kind.

**Proposition A.1.** *For all  $N \geq 1$ ,*

- *for  $k = 0, \dots, N-1$ , the polynomials  $T_k, -T_{2N-k}, -T_{2N+k}, T_{4N-k}, T_{4N+k}, \dots$  take the same values at the  $\mu_{j,N-1}$ ,  $j = 0, \dots, N-1$ ,*
- *for  $j \geq 0$ , let*

$$m = |(j + N - 1)(\text{mod } 2N) - N + 1| \text{ and } p = \left\lfloor \frac{N + j}{2N} \right\rfloor,$$

*the polynomials  $T_j$  and  $(-1)^p T_m$  take the same values at the  $\mu_{j,N-1}$ ,  $j = 0, \dots, N-1$ .*

*Proof.* These are Theorems 1 and 2 of [69]. □

Let  $f$  be Lipschitz continuous over  $[-1, 1]$ , we know [71, Chap. VI] that  $f$  admits a series expansion  $\sum'_{n \geq 0} a_n T_n(x)$  in  $L_2([-1, 1], (1 - x^2)^{-1/2})$  which converges uniformly to  $f$ . We now state a straightforward consequence of Proposition A.1.

**Corollary A.2.** *Let  $N \in \mathbb{N}$ ,  $N \geq 1$ , Let  $f$  be Lipschitz continuous over  $[-1, 1]$ , its Chebyshev coefficients  $(a_k)_{k \geq 0}$  and the coefficients  $c_k$ ,  $k = 0, \dots, N-1$ , of the interpolation polynomial  $p_{N-1}$  of  $f$  at the Chebyshev nodes of the first kind satisfy, for  $k = 0, \dots, N-1$ ,*

$$(A.1) \quad \begin{aligned} c_k &= a_k - a_{2N-k} - a_{2N+k} + a_{4N-k} + a_{4N-k} - \dots \\ &= \sum_{j=0}^{+\infty} (-1)^j a_{2jN+k} + \sum_{j=1}^{+\infty} (-1)^j a_{2jN-k}. \end{aligned}$$

Let  $N \in \mathbb{N}$ ,  $N \geq 1$ , we also define

$$\gamma_{\rho,0,N-1} = 1 \text{ and } \gamma_{\rho,k,N-1} = \frac{1}{1 - \rho^{-2N}} \left( 1 + \frac{1}{\rho^{2(N-k)}} \right) \text{ for } k = 1, \dots, N-1.$$

**Proposition A.3.** *Let  $\rho > 1$ , let  $N \in \mathbb{N}$ ,  $N \geq 1$ ,  $f$  be a function analytic in a neighbourhood of  $E_\rho$ , the coefficients  $(c_k)_{k=0,\dots,N-1}$  of the interpolation polynomial of  $f$  at the Chebyshev nodes of the first kind satisfy*

$$|c_k| \leq 2 \frac{M_\rho(f)}{\rho^k} \gamma_{\rho,k,N-1}, \quad k = 0, \dots, N-1,$$

where  $M_\rho(f) = \max_{z \in E_\rho} |f(z)|$ . Moreover, we have

$$\|f - p_{N-1}\|_{\infty,[-1,1]} \leq \frac{4M_\rho(f)}{\rho^{N-1}(\rho - 1)}.$$

*Proof.* First, we use the following consequence of [65, Thm 8.1]: the Chebyshev coefficients satisfy

$$(A.2) \quad |a_0| \leq M_\rho(f) \text{ and } |a_k| \leq 2 \frac{M_\rho(f)}{\rho^k}.$$

Then, we combine Equation (A.1) and Inequalities (A.2) to obtain, for  $k = 1, \dots, N-1$ .

$$\begin{aligned} |c_k| &\leq |a_k| + |a_{2N-k}| + |a_{2N+k}| + |a_{4N-k}| + |a_{4N+k}| + \dots, \\ &\leq 2 \frac{M_\rho(f)}{\rho^k} + 2 \frac{M_\rho(f)}{\rho^{2N-k}} + 2 \frac{M_\rho(f)}{\rho^{2N+k}} + 2 \frac{M_\rho(f)}{\rho^{4N-k}} + 2 \frac{M_\rho(f)}{\rho^{4N+k}} + \dots, \\ &\leq 2 \frac{M_\rho(f)}{\rho^k} \left( 1 + \frac{1}{\rho^{2N-2k}} + \frac{1}{\rho^{2N}} + \frac{1}{\rho^{4N-2k}} + \frac{1}{\rho^{4N}} + \dots \right), \\ &\leq 2 \frac{M_\rho(f)}{\rho^k} \frac{1}{1 - \rho^{-2N}} \left( 1 + \frac{1}{\rho^{2(N-k)}} \right). \end{aligned}$$

Moreover, recall that  $c_0 = \frac{2}{N} \sum_{1 \leq \ell \leq N} f(\mu_\ell)$ , hence  $|c_0| = 2 \max_{x \in [-1, 1]} |f(x)| \leq 2M_\rho(f)$  by the maximum principle.

Now, we turn to the estimate on the remainder. Corollary A.2 yields, for any  $x \in [-1, 1]$ ,

$$f(x) - p_{N-1}(x) = \sum_{k \geq N} a_k (T_k(x) - (-1)^k T_m(x))$$

where  $m$  and  $p$  are defined as in Proposition A.1. Hence, we have, for any  $x \in [-1, 1]$ ,

$$\begin{aligned} |f(x) - p_{N-1}(x)| &\leq \sum_{k \geq N} |a_k| |T_k(x) - (-1)^k T_m(x)| \\ &\leq 2 \sum_{k \geq N} |a_k| \leq 4M_\rho(f) \sum_{k \geq N} \rho^{-k} = \frac{4M_\rho(f)}{\rho^{N-1}(\rho-1)}. \end{aligned}$$

□

Regarding the two variable case, we start by establishing results analogous to [65, Thms 8.1 and 8.2]. Let  $f$  in  $L_2([-1, 1] \times [-1, 1], (1-x^2)^{-1/2}(1-y^2)^{-1/2})$ , we denote by  $\sum'_{n_1 \geq 0} \sum'_{n_2 \geq 0} a_{n_1, n_2} T_{n_1}(x) T_{n_2}(y)$  its series expansion.

**Proposition A.4.** *Let  $\rho_1, \rho_2 > 1$ ,  $f$  be a function analytic in a neighbourhood of  $E_{\rho_1, \rho_2}$ , the coefficients  $a_{n_1, n_2}, n_1, n_2 \geq 0$ , of the Chebyshev series of  $f$  satisfy, for all  $n_1$  and  $n_2 \in \mathbb{N}$ ,*

$$(A.3) \quad |a_{n_1, n_2}| \leq 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{n_1} \rho_2^{n_2}},$$

where  $M_{\rho_1, \rho_2}(f) = \max_{z \in \mathcal{E}_{\rho_1, \rho_2}} |f(z)|$ . Moreover, we have, for all  $n_1$  and  $n_2 \in \mathbb{N}$ ,

$$\begin{aligned} \left\| f(x, y) - \sum'_{k_1=0}^{n_1} \sum'_{k_2=0}^{n_2} a_{k_1, k_2} T_{k_1}(x) T_{k_2}(y) \right\|_{\infty, [-1, 1] \times [-1, 1]} &\leq \frac{4\rho_1 \rho_2 M_{\rho_1, \rho_2}(f)}{(\rho_1 - 1)(\rho_2 - 1)} \left( \frac{1}{\rho_1^{n_1+1}} + \frac{1}{\rho_2^{n_2+1}} \right). \end{aligned}$$

*Proof.* For  $\rho > 0$ , we define  $\mathcal{C}_\rho = \{z \in \mathbb{C}, |z| = \rho\}$ . Extending what is done in the proof of [65, Thm. 8.1], we now introduce the change of variables  $x = (z_1 + z_1^{-1})/2, y = (z_2 + z_2^{-1})/2$  where  $z_1, z_2 \in \mathcal{C}_1$  and the function

$$f(x, y) = F(z_1, z_2) = \sum'_{n_1 \geq 0} \sum'_{n_2 \geq 0} a_{n_1, n_2} \frac{z_1^{n_1} + z_1^{-n_1}}{2} \frac{z_2^{n_2} + z_2^{-n_2}}{2}.$$

Now we use Cauchy's integral formula in two variables: for all  $n_1, n_2 \in \mathbb{N}$ ,

$$\frac{1}{(2i\pi)^2} \int_{\mathcal{C}_1 \times \mathcal{C}_1} \frac{F(z_1, z_2)}{z_1^{n_1+1} z_2^{n_2+1}} dz_1 dz_2 = \frac{1}{2^{\delta_{0n_1} + \delta_{0n_2}}} \frac{2^{\delta_{0n_1} + \delta_{0n_2}}}{4} a_{n_1, n_2}.$$

If  $g : (z_1, z_2) \mapsto ((z_1 + z_1^{-1})/2, (z_2 + z_2^{-1})/2)$ , the domain  $E_{\rho_1, \rho_2}$  is the image of  $\mathcal{R}_{\rho_1} \times \mathcal{R}_{\rho_2}$ , where  $\mathcal{R}_\rho = \{z \in \mathbb{C}, \rho^{-1} < |z| < \rho\}$ , via the application  $g$ . Note that, since  $F = f \circ g$ ,  $F$  is analytic in a neighbourhood of  $\mathcal{R}_{\rho_1} \times \mathcal{R}_{\rho_2}$  since it is the composition of two analytic functions, hence, for all  $n_1, n_2 \in \mathbb{N}$ ,

$$a_{n_1, n_2} = \frac{1}{(i\pi)^2} \int_{\mathcal{C}_{\rho_1} \times \mathcal{C}_{\rho_2}} \frac{F(z_1, z_2)}{z_1^{n_1+1} z_2^{n_2+1}} dz_1 dz_2,$$

from which follows

$$|a_{n_1, n_2}| \leq \frac{(2\pi)^2 \rho_1 \rho_2 \max_{(z_1, z_2) \in \mathcal{C}_{\rho_1} \times \mathcal{C}_{\rho_2}} |F(z_1, z_2)|}{\pi^2 \frac{\rho_1^{n_1+1} \rho_2^{n_2+1}}{\rho_1^{n_1} \rho_2^{n_2}}} = \frac{4M_{\rho_1, \rho_2}(f)}{\rho_1^{n_1} \rho_2^{n_2}} \text{ for all } n_1, n_2 \in \mathbb{N}.$$

As for the remainder, for all  $x, y \in [-1, 1]$ , for all  $n_1$  and  $n_2 \in \mathbb{N}$ , we have

$$\begin{aligned} f(x, y) - \sum_{k_1=0}^{n_1'} \sum_{k_2=0}^{n_2'} a_{k_1, k_2} T_{k_1}(x) T_{k_2}(y) \\ = \sum_{k_1 \geq n_1+1} \sum_{k_2 \geq 0} a_{k_1, k_2} T_{k_1}(x) T_{k_2}(y) + \sum_{k_1=0}^{n_1'} \sum_{k_2 \geq n_2+1} a_{k_1, k_2} T_{k_1}(x) T_{k_2}(y), \end{aligned}$$

hence,

$$\begin{aligned} \left\| f(x, y) - \sum_{k_1=0}^{n_1'} \sum_{k_2=0}^{n_2'} a_{k_1, k_2} T_{k_1}(x) T_{k_2}(y) \right\|_{\infty, [-1, 1] \times [-1, 1]} \\ = \sum_{k_1 \geq n_1+1} \sum_{k_2 \geq 0} |a_{k_1, k_2}| + \sum_{k_1=0}^{n_1'} \sum_{k_2 \geq n_2+1} |a_{k_1, k_2}|, \\ \leq 4M_{\rho_1, \rho_2}(f) \left( \frac{1}{\rho_1^{n_1+1}} + \frac{1}{\rho_2^{n_2+1}} \right) \frac{\rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)}. \end{aligned}$$

□

Now we can prove:

**Proposition A.5.** *Let  $\rho_1, \rho_2 > 1$ , let  $M_1, M_2 \in \mathbb{N}$ ,  $M_1, M_2 \geq 2$ ,  $f$  be a function analytic in a neighbourhood of  $E_{\rho_1, \rho_2}$ , the coefficients  $c_{k_1, k_2}$ ,  $k_1 = 0, \dots, M_1 - 1$ ,  $k_2 = 0, \dots, M_2 - 1$  of the interpolation polynomial  $P_{M_1-1, M_2-1}$  of  $f$  at pairs of Chebyshev nodes of the first kind satisfy, for  $k_1 = 1, \dots, M_1 - 1$ ,  $k_2 = 1, \dots, M_2 - 1$ ,*

$$|c_{k_1, k_2}| \leq 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{k_1} \rho_2^{k_2}} \gamma_{\rho_1, k_1, M_1-1} \gamma_{\rho_2, k_2, M_2-1},$$

where  $M_{\rho_1, \rho_2}(f) = \max_{z \in \mathcal{E}_{\rho_1, \rho_2}} |f(z)|$ . Moreover, we have

$$\|f - P_{M_1-1, M_2-1}\|_{\infty, [-1, 1] \times [-1, 1]} \leq \frac{16\rho_1 \rho_2 M_{\rho_1, \rho_2}(f)}{(\rho_1 - 1)(\rho_2 - 1)} \left( \frac{1}{\rho_1^{M_1}} + \frac{1}{\rho_2^{M_2}} \right).$$

*Proof.* Let  $\sum'_{n_1 \geq 0} \sum'_{n_2 \geq 0} a_{n_1, n_2} T_{n_1}(x) T_{n_2}(y)$  the series expansion of  $f$ , the aliasing phenomenon presented above still exists: for  $k_1 = 0, \dots, M_1 - 1$ ,  $k_2 = 0, \dots, M_2 - 1$ ,

$$(A.4) \quad c_{k_1, k_2} = \sum_{p_1=0}^{+\infty} \sum_{p_2=0}^{+\infty} (-1)^{p_1+p_2} a_{2p_1 M_1+k_1, 2p_2 M_2+k_2} \\ + \sum_{p_1=0}^{+\infty} \sum_{p_2=1}^{+\infty} (-1)^{p_1+p_2} a_{2p_1 M_1+k_1, 2p_2 M_2-k_2} + \sum_{p_1=1}^{+\infty} \sum_{p_2=0}^{+\infty} (-1)^{p_1+p_2} a_{2p_1 M_1-k_1, 2p_2 M_2+k_2} \\ + \sum_{p_1=1}^{+\infty} \sum_{p_2=1}^{+\infty} (-1)^{p_1+p_2} a_{2p_1 M_1-k_1, 2p_2 M_2-k_2}.$$

We now combine Equation (A.4) and Inequalities (A.3) to obtain, for  $k_1 = 1, \dots, M_1 - 1$ ,  $k_2 = 1, \dots, M_2 - 1$ ,

$$|c_{k_1, k_2}| \leq \sum_{p_1=0}^{+\infty} \sum_{p_2=0}^{+\infty} |a_{2p_1 M_1+k_1, 2p_2 M_2+k_2}| + \sum_{p_1=0}^{+\infty} \sum_{p_2=1}^{+\infty} |a_{2p_1 M_1+k_1, 2p_2 M_2-k_2}| \\ + \sum_{p_1=1}^{+\infty} \sum_{p_2=0}^{+\infty} |a_{2p_1 M_1-k_1, 2p_2 M_2+k_2}| + \sum_{p_1=1}^{+\infty} \sum_{p_2=1}^{+\infty} |a_{2p_1 M_1-k_1, 2p_2 M_2-k_2}| \\ \leq \sum_{p_1=0}^{+\infty} \sum_{p_2=0}^{+\infty} 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{2p_1 M_1+k_1} \rho_2^{2p_2 M_2+k_2}} + \sum_{p_1=0}^{+\infty} \sum_{p_2=1}^{+\infty} 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{2p_1 M_1+k_1} \rho_2^{2p_2 M_2-k_2}} \\ + \sum_{p_1=1}^{+\infty} \sum_{p_2=0}^{+\infty} 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{2p_1 M_1-k_1} \rho_2^{2p_2 M_2+k_2}} + \sum_{p_1=1}^{+\infty} \sum_{p_2=1}^{+\infty} 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{2p_1 M_1-k_1} \rho_2^{2p_2 M_2-k_2}} \\ \leq 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{k_1} \rho_2^{k_2}} \frac{1}{1 - \rho_1^{-2M_1}} \frac{1}{1 - \rho_2^{-2M_2}} \\ \left( 1 + \frac{1}{\rho_1^{2(M_1-k_1)}} + \frac{1}{\rho_2^{2(M_2-k_2)}} + \frac{1}{\rho_1^{2(M_1-k_1)} \rho_2^{2(M_2-k_2)}} \right).$$

If we use Equation (6.1), we get

$$|c_{0,0}| \leq 4M_{\rho_1, \rho_2}(f) \text{ thanks to the maximum principle,} \\ |c_{k_1,0}| \leq 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_1^{k_1}} \frac{1}{1 - \rho_1^{-2M_1}} \left( 1 + \frac{1}{\rho_1^{2(M_1-k_1)}} \right) \text{ for } k_1 = 1, \dots, M_1 - 1, \\ |c_{0,k_2}| \leq 4 \frac{M_{\rho_1, \rho_2}(f)}{\rho_2^{k_2}} \frac{1}{1 - \rho_2^{-2M_2}} \left( 1 + \frac{1}{\rho_2^{2(M_2-k_2)}} \right) \text{ for } k_2 = 1, \dots, M_2 - 1.$$

The last two inequalities are consequences of Proposition A.3.

As for the remainder, for all  $x, y \in [-1, 1]$ , for all  $n_1$  and  $n_2 \in \mathbb{N}$ , we have thanks to the aliasing phenomenon

$$\begin{aligned} f(x, y) &- \sum'_{k_1=0}^{M_1-1} \sum'_{k_2=0}^{M_2-1} c_{k_1, k_2} T_{k_1}(x) T_{k_2}(y) \\ &= \sum'_{k_1 \geq M_1} \sum'_{k_2 \geq 0} a_{k_1, k_2} (T_{k_1}(x) - (-1)^{p_1} T_{m_1}(x)) (T_{k_2}(y) - (-1)^{p_2} T_{m_2}(y)) \\ &\quad + \sum'_{k_1=0}^{M_1-1} \sum'_{k_2 \geq M_2} a_{k_1, k_2} (T_{k_1}(x) - (-1)^{p_1} T_{m_1}(x)) (T_{k_2}(y) - (-1)^{p_2} T_{m_2}(y)), \end{aligned}$$

where  $m_1, m_2$  and  $p_1, p_2$  are defined as in Proposition A.1. Hence,

$$\begin{aligned} \|f(x, y) - P_{M_1-1, M_2-1}(x, y)\|_{\infty, [-1, 1] \times [-1, 1]} &= \sum'_{k_1 \geq M_1} \sum'_{k_2 \geq 0} 4|a_{k_1, k_2}| \\ &\quad + \sum'_{k_1=0}^{M_1-1} \sum'_{k_2 \geq M_2} 4|a_{k_1, k_2}| \leq 16M_{\rho_1, \rho_2}(f) \left( \frac{1}{\rho_1^{M_1}} + \frac{1}{\rho_2^{M_2}} \right) \frac{\rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)}, \end{aligned}$$

thanks to Proposition A.4.  $\square$

The next lemma eases the computations.

**Lemma A.6.** *Let  $\rho > 1$ ,  $N \geq 2$ , for  $k = 1, \dots, N-1$ , we have*

$$\gamma_{\rho, k, N-1} \leq \frac{\rho^2 + 1}{\rho^2 - 1}.$$

*In particular, if  $\rho \geq 2$ ,  $\gamma_{\rho, k, N-1} \leq 2$  for  $k = 1, \dots, N-1$ .*

*Proof.* For  $k = 1, \dots, N-1$ , we have

$$\gamma_{\rho, k, N-1} = \frac{1}{1 - \rho^{-2N}} \left( 1 + \frac{1}{\rho^{2(N-k)}} \right) = \frac{1}{\rho^{2N} - 1} (\rho^{2N} + \rho^{2k}) \leq \frac{\rho^{2N}}{\rho^{2N} - 1} (1 + \rho^{-2}).$$

The function  $u \mapsto u/(u-1)$  is strictly decreasing over  $(1, +\infty)$ , hence

$$\gamma_{\rho, k, N-1} \leq \frac{\rho^{2N}}{\rho^{2N} - 1} (1 + \rho^{-2}) \leq \frac{\rho^2}{\rho^2 - 1} (1 + \rho^{-2}) = \frac{\rho^2 + 1}{\rho^2 - 1}. \quad \square$$

## APPENDIX B. PROOF OF THEOREM 5.24

For  $f(z) = \exp(\exp(z))$ , we take  $K = \log d$ . A sufficient condition for success of Algorithm 2 is, in view of Proposition 5.11, (for  $d$  large enough)

$$\begin{aligned} b - a &< \frac{1}{4} \left( (|a + b| + \log d) M_{\mathcal{D}_{a, b, K}}(f) \right)^{-4/(3(1-\lambda^2)d(1+o(1)))} \\ &\quad (uv)^{-4/(3(1-\lambda^2)d(1+o(1)))} \log d. \end{aligned}$$

We have  $M_{\mathcal{D}_{a, b, K}}(f) = 2^{O(d)}$ ; hence, as in the proof of Theorem 5.23, a sufficient condition is

$$(uv)^{-4/(3(1-\lambda^2)d)} \log d \rightarrow \infty,$$

for which it suffices that

$$d \log \log d \geq \left( \frac{4}{3(1-\lambda^2)} + \varepsilon' \right) \log(uv),$$

for some  $\varepsilon' > 0$ , which follows from the assumption of the Theorem. The statement on  $w$  follows from simple calculus using  $w = O(K^{N(1-\lambda)})$ .

For  $g(z)$ , we have

$$\begin{aligned} \sup_{|z|=\rho} |g(z)| &= g(\rho) = \exp(\log^2 \rho/4) \sum_{n \geq 0} \exp(-(n - \log \rho/2)^2) \\ &\leq \exp(\log^2 \rho/4) \sum_{n \in \mathbb{Z}} \exp(-n^2) \leq 2 \exp(\log^2 \rho/4). \end{aligned}$$

We choose  $K = \exp(3(1-\lambda^2)d/2)$ . In this case, we have  $M_{\mathcal{D}_{a,b,K}}(f) = \exp(9(1-\lambda^2)^2 d^2/16 + O(d))$ .

A sufficient success condition for Algorithm 2 is thus, as  $d \rightarrow \infty$ ,

$$\begin{aligned} b - a &< \frac{1}{4} (|a + b| + \exp(3(1-\lambda^2)d/2))^{-4/(3(1-\lambda^2)d(1+o(1)))} \\ &\quad \exp(3(1-\lambda^2)d/2) (uv M_{\mathcal{D}_{a,b,K}}(f))^{-4/(3(1-\lambda^2)d(1+o(1)))}. \end{aligned}$$

It follows from

$$\frac{3(1-\lambda^2)d}{2} - \frac{4(1+o(1))}{3(1-\lambda^2)d} \left( \log(uv) + \frac{9(1-\lambda^2)^2 d^2}{16} \right) \rightarrow \infty,$$

for which it suffices to have

$$d^2 \geq \left( \frac{16}{9(1-\lambda^2)^2} + \varepsilon' \right) \log(uv),$$

for some  $\varepsilon' > 0$ .

Again, the statement on  $w$  follows from simple calculus using  $w = O(K^{N(1-\lambda)})$ .

### APPENDIX C. LEMMATA ON $\varphi, \psi$

In this appendix, we group the facts concerning the function  $\psi$  of Section 6.

**Lemma C.1.** *Let  $\varphi$  be the function from  $[1, +\infty)$  to  $[1, +\infty)$  defined by  $\varphi(x) = (1 + [x])(x - [x])/2$ . Then  $\varphi$  is continuous and strictly increasing, and defines a bijection from  $[1, +\infty)$  to  $[1, +\infty)$ . For any  $x \geq 1$ , we have  $x(x+1)/2 \leq \varphi(x) \leq (x+1/2)^2/2$  and  $\sqrt{2x} - 1/2 \leq \varphi^{-1}(x) \leq \sqrt{2x+1/4} - 1/2$ .*

*Proof.* For  $x \notin \mathbb{Z}$ , it is clear that  $\varphi$  is  $C^1$  in a neighbourhood of  $x$  and that  $\varphi'(x) \geq 1$ .

For  $x$  in  $\mathbb{Z}$ , we have  $\varphi(x) = \lim_{t \rightarrow x^+} \varphi(t) = (1+x)x/2$ , whereas  $\lim_{t \rightarrow x^-} \varphi(t) = x(x - (x-1)/2) = x(x+1)/2$ . This proves continuity, and the remaining assertions follow.

Let  $k \in \mathbb{N}, a \in \mathbb{R}$  and  $g_a(x) = (x+1-a)(x+a)/2$ . We denote by  $\varphi_k$  the restriction of  $\varphi$  to  $[k, k+1]$ . For all  $x \in [k, k+1]$ ,  $(\varphi_k - g_a)'(x) = k+1/2 - x$ : the function  $\varphi_k - g_a$  is decreasing over  $[k, k+1/2]$  and increasing over  $[k+1/2, k+1]$ . Now, we remark that  $\varphi_k(k) = g_0(k)$  and  $\varphi_k(k+1) = g_0(k+1)$ , which yields  $\varphi_k(x) \geq g_0(x) = x(x+1)/2$  for all  $x \in [k, k+1]$ , and  $\varphi_k(k+1/2) = g_{1/2}(k+1/2)$ , which yields  $\varphi_k(x) \leq g_{1/2}(x) = (x+1/2)^2/2$  for all  $x \in [k, k+1]$ .

The proof of the remaining inequalities is straightforward.  $\square$

**Lemma C.2.** *Let  $3 \leq \gamma \leq N$ ,  $s = \gamma\varphi^{-1}(N/\gamma)$ ,  $N_1 = \lfloor \sqrt{2N\gamma} \rfloor$  and  $N_2 = \lceil \sqrt{2N/\gamma} \rceil$ . We have  $\gamma \geq N_1/N_2 \geq 1$ ,  $s < N_1$  and  $\sqrt{2N}(\sqrt{2N} - \sqrt{3}/3) \leq N_1N_2 \leq (2 + \sqrt{2})N$ .*

*Assume that  $N \rightarrow \infty$ , then  $\text{card } \mathcal{K}_s \leq N + O(s/\gamma)$ .*

*Proof.* We have

$$\sqrt{2N\gamma} - \sqrt{2N/\gamma} = \sqrt{2N\gamma}(1 - 1/\gamma) \geq 1,$$

since  $N \geq \gamma \geq 3$ . It follows  $N_1 = \lfloor \sqrt{2N\gamma} \rfloor \geq \lceil \sqrt{2N/\gamma} \rceil = N_2$ , hence  $N_1/N_2 \geq 1$ . Moreover, for any  $\gamma \geq 3$ ,

$$N_1 = \lfloor \sqrt{2N\gamma} \rfloor \leq \gamma\sqrt{2N/\gamma} \leq \gamma N_2.$$

Also,

$$(\sqrt{2N\gamma} - 1)\sqrt{2N/\gamma} \leq N_1N_2 \leq \sqrt{2N\gamma}(1 + \sqrt{2N/\gamma}),$$

hence

$$\sqrt{2N}(\sqrt{2N} - \sqrt{3}/3) \leq N_1N_2 \leq (2 + \sqrt{2})N$$

since  $N \geq \gamma \geq 2$ .

Finally, from Lemma C.1 and the fact that  $(\sqrt{8x+1} - 1)/2 < \sqrt{2x} - 3/8$  for all  $x \geq 1$ , we have  $s = \gamma\varphi^{-1}(N/\gamma) < \gamma(\sqrt{2N/\gamma} - 3/8) < \sqrt{2N\gamma} - 1 \leq N_1$  since  $N \geq \gamma \geq 3$ . Therefore, we can apply Lemma 6.3: we have, from (6.5),

$$\text{card } \mathcal{K}_s = \gamma(1 + \lfloor s/\gamma \rfloor)(s/\gamma - \lfloor s/\gamma \rfloor/2) + O(s/\gamma) \leq \gamma\varphi(s/\gamma) + O(s/\gamma) \leq N + O(s/\gamma).$$

□

**Corollary C.3.** *With the assumptions of Lemma C.2, put  $\lambda = N/\gamma$ . Then, we have*

$$\Omega_\gamma(N, N_1, N_2) = \psi(\lambda)N\gamma + O(N),$$

where

$$\psi(\lambda) = \frac{1 + \lfloor \varphi^{-1}(\lambda) \rfloor}{12\lambda} (6\varphi^{-1}(\lambda)^2 - \lfloor \varphi^{-1}(\lambda) \rfloor - 2\lfloor \varphi^{-1}(\lambda) \rfloor^2).$$

*Note that if  $\gamma = o(N)$ , we have  $\lambda \rightarrow \infty$  and  $\varphi^{-1}(\lambda) = \sqrt{2\lambda} + O(1)$ , so that  $\psi(\lambda) = 2\sqrt{2\lambda}/3 + O(1)$  and*

$$\Omega_\gamma(N, N_1, N_2) = \frac{2\sqrt{2}}{3}N^{3/2}\gamma^{1/2} + O(N\gamma).$$

*Proof.* Lemma C.2 shows us that the assumptions of Lemma 6.3 are satisfied; we thus get

$$\sum_{(i,j) \in \mathcal{K}_s} (i + j\gamma) = \psi(\lambda)N\gamma + O(N).$$

Further, note that for our value of  $s$ , the term  $s(N - \text{card } \mathcal{K}_s)$  from Lemma 6.1 is  $O(s^2/\gamma)$ , which is  $O(\gamma\varphi^{-1}(\lambda)^2) = O(N\varphi^{-1}(\lambda)^2/\lambda) = O(N)$ , thanks to Lemma C.1. The result follows. □

**Lemma C.4.** *For  $x \in [1, +\infty)$ , we have*

$$-5/6 < \psi(x) - \frac{2\sqrt{2x}}{3} < 0.$$

*Proof.* We have

$$\psi(\varphi(x)) = \frac{6x^2 - \lfloor x \rfloor - 2\lfloor x \rfloor^2}{12x - 6\lfloor x \rfloor}.$$

For  $v \leq u < v + 1$ , define

$$F(u, v) = \frac{6u^2 - v - 2v^2}{12u - 6v} - 2u/3,$$

so that  $\psi(\varphi(x)) - 2x/3 = F(x, \lfloor x \rfloor)$ .

We maximize  $F(u, v)$  for fixed  $v$ , hence computing

$$\frac{\partial F}{\partial v}(u, v) = \frac{-2u^2 + 2uv + v}{3(4u^2 - 4uv + v^2)}.$$

By evaluating  $2u^2 - 2uv - v = 0$  at  $v$  and  $v + 1$ , one checks that for fixed  $v$ , there is a unique  $u_0 \in [v, v + 1)$  such that  $F(u, v)$  increases over  $[v, u_0]$  and decreases over  $[u_0, v + 1)$ . Hence, for  $u \in [v, v + 1)$ ,

$$\frac{-1}{6} = \min(F(v, v) - 2v/3, F(v+1, v) - 2(v+1)/3) \leq F(u, v) - 2u/3 \leq F(u_0, v) - 2u_0/3.$$

Finally, we find

$$F(u_0, v) - \frac{2u_0}{3} = (v - u_0)/3 \leq 0.$$

(note that the optimal bound for the latter is actually  $(1 - \sqrt{3})/6$ , obtained for  $v = 1$ ,  $u_0 = (1 + \sqrt{3})/2$ ).

Hence, we have

$$2\varphi^{-1}(x)/3 - 1/6 \leq \psi(x) \leq 2\varphi^{-1}(x)/3,$$

which, in view of  $\sqrt{2x} - 1 < \varphi^{-1}(x) < \sqrt{2x}$ , gives

$$-5/6 < \psi(x) - 2\sqrt{2x}/3 < 0.$$

□

*Remark C.5.* Simple numerical experiments suggest that actually  $\psi(x) - 2\sqrt{2x}/3 \in [-1/2, -0.44]$  for  $x \geq 1$ , so that the asymptotic expansion  $\psi(x) = 2\sqrt{2x}/3 - 1/2 + o(1)$ , once truncated, actually gives an excellent approximation for all  $x \geq 1$ .

The following two lemmas yield useful information on the function  $\psi$ : invertibility, and inverse function.

**Lemma C.6.** *The function  $x \mapsto \psi(x)$  is continuous and increasing over  $[1, \infty)$ .*

*Proof.* For the first part, since  $\varphi^{-1}$  is continuous and  $x > 0$ , it suffices to prove that  $x \mapsto x(\psi \circ \varphi(x))$  is continuous, namely that  $F : x \mapsto (1 + \lfloor x \rfloor)(6x^2 - \lfloor x \rfloor - 2\lfloor x \rfloor^2)$  is continuous.

Obviously,  $F$  is continuous on  $[1, \infty) \setminus \mathbb{Z}_{>0}$ . If  $n$  is an integer, we check that  $F(n) = (1+n)(4n^2 - n)$  whereas  $\lim_{x \rightarrow n^-} F(x) = n(4n^2 + 3n - 1) = n(n+1)(4n - 1) = F(n)$ .

To prove that  $x \mapsto \psi(x)$  is increasing, as  $\varphi$  is increasing it suffices to study  $x \mapsto \psi(\varphi(x))$ . As the latter function is continuous over each interval  $(\varphi^{-1}(n), \varphi^{-1}(n+1))$ , it suffices to prove that it increases over each of those intervals.

Over such an interval, we have  $\psi(\varphi(x)) = Ax^2/\varphi(x) - B/\varphi(x)$  for some nonnegative constants  $A, B$ ; it thus suffices to prove that  $x^2/\varphi(x)$  is increasing, or that  $\varphi(x)/x^2$  is decreasing. As this function is continuous and has the form  $A'(x+B')/x^2$  for some nonnegative  $A', B'$  over each interval  $(n, n+1)$  for integer  $n$ , we see that it is indeed decreasing. □



**Proposition C.7.** *We have, for any  $\mu \in [1, +\infty)$ ,*

$$\psi^{-1}(\mu) = \frac{k+1}{2} \left( 2\mu - k + \sqrt{4\mu(\mu - k) + 2k(2k+1)/3} \right),$$

where  $k = \lfloor 3\mu/2 + 1/4 \rfloor$ . For  $\mu \rightarrow \infty$ , we have  $\psi^{-1}(\mu) = 9\mu^2/8 + O(\mu)$ .

*Proof.* We start by noticing that for all integer  $\ell$ ,  $\psi(\ell(\ell+1)/2) = (4\ell-1)/6$ , which follows easily from  $\varphi^{-1}(\ell(\ell+1)/2) = \ell$ .

Let now  $k = \lfloor 3\mu/2 + 1/4 \rfloor$ ; then,  $\psi(k(k+1)/2) \leq \mu < \psi((k+1)(k+2)/2)$ , so that

$$k(k+1)/2 \leq \psi^{-1}(\mu) < (k+1)(k+2)/2,$$

and  $\lfloor \varphi^{-1}(\psi^{-1}(\mu)) \rfloor = k$ . The asymptotic expansion follows from these inequalities.

As a consequence,  $\psi^{-1}(\mu) = \varphi(\varphi^{-1}(\psi^{-1}(\mu))) = (k+1)(\varphi^{-1}(\psi^{-1}(\mu)) - k/2)$ , so that  $\varphi^{-1}(\psi^{-1}(\mu)) = \frac{\psi^{-1}(\mu)}{k+1} + k/2$ .

Hence,

$$\mu = \psi(\psi^{-1}(\mu)) = \frac{1+k}{12\psi^{-1}(\mu)} (6\varphi^{-1}(\psi^{-1}(\mu))^2 - k - 2k^2),$$

so that

$$\frac{12}{1+k} \mu \psi^{-1}(\mu) = 6 \left( \frac{\psi^{-1}(\mu)}{k+1} + k/2 \right)^2 - k - 2k^2.$$

Finally,  $\psi^{-1}(\mu)/(k+1)$  is the positive root of the equation

$$X^2 + (k - 2\mu)X - \frac{k(k+2)}{12} = 0,$$

which gives the explicit form

$$\psi^{-1}(\mu) = \frac{k+1}{2} \left( 2\mu - k + \sqrt{4\mu(\mu - k) + 2k(2k+1)/3} \right).$$

□

#### APPENDIX D. PROOFS OF THEOREMS 6.4 AND 6.13

*Proof of Theorem 6.4.* For  $0 \leq i, j \leq N-1$ ,  $0 \leq k_1 \leq N_1-1$ ,  $0 \leq k_2 \leq N_2-1$ , we have from Proposition 4.2 and Lemma A.6,

$$\begin{aligned} |(A_1)_{i,(k_1,k_2)}| &\leq \left| \frac{C_{k_1,k_2,i}}{2^{\delta_{0k_1} + \delta_{0k_2}}} \right| \leq \\ &4 \frac{M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{\rho_1^{k_1} \rho_2^{k_2}} \frac{\rho_1^2 + 1}{\rho_1^2 - 1} \frac{\rho_2^2 + 1}{\rho_2^2 - 1} \leq \frac{4\rho_1\rho_2 M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{(\rho_1 - 1)(\rho_2 - 1)} \frac{1}{\rho_1^{k_1} \rho_2^{k_2}}. \end{aligned}$$

Moreover, from the definition of  $A_2$  and the assumption  $\rho_1^{N_1} \leq \rho_2^{N_2}$ , we know that, for  $0 \leq i, j \leq N-1$ ,

$$|A_{2,i,j}| \leq \frac{16\rho_1\rho_2 M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{(\rho_1 - 1)(\rho_2 - 1)} \left( \frac{2}{\rho_1^{N_1}} \right).$$

We now apply Theorem 5.1. We put  $r_i = \frac{32\rho_1\rho_2 M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{(\rho_1 - 1)(\rho_2 - 1)}$  for  $i = 0, \dots, N-1$ . Then, notice that the product of the  $N$  largest elements among

$$1, \dots, \frac{1}{\rho_1^{k_1} \rho_2^{k_2}}, \dots, \frac{1}{\rho_1^{N_1-1} \rho_2^{N_2-1}}, \underbrace{\frac{1}{\rho_1^{N_1}}, \dots, \frac{1}{\rho_1^{N_1}}}_{N \text{ times}}.$$

Thanks to the assumption  $\rho_1^\gamma = \rho_2$ , or equivalently,  $\rho_1^{-i}\rho_2^{-j} = \rho_1^{-i-\gamma j}$ , this product is upper bounded by  $\rho_1^{-\Omega_\gamma(N_1, N_2, N)}$ .

We can now apply Theorem 5.1 to obtain the bound

$$(\det A^t A)^{1/2} \leq \left(32\sqrt{N}\right)^N 2^{\frac{N_1 N_2 + N}{2}} \left(\frac{\rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)}\right)^N \frac{\prod_{i=1}^N M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f_i)}{\rho_1^{\Omega_\gamma(N, N_1, N_2)}},$$

from which the Theorem follows, in view of the preliminary assumption that  $N_1 N_2 \geq N$ .  $\square$

*Proof of Theorem 6.13.* The assumption  $\|\Lambda \hat{A}\|_2 < 1/(N + N_1 N_2)$  implies in particular, for all  $k, \ell$ ,

$$1/(N + N_1 N_2) \geq |\lambda_{k, \ell}| \min_i \hat{A}_2[i, i] \geq 2^{1-\text{prec}} N |\lambda_{k, \ell}|,$$

cf. proof of Lemma 6.10, hence  $|\lambda_{k, \ell}| \leq 2^{\text{prec}-1}/(N(N + N_1 N_2))$ . For  $j = 0, \dots, N + N_1 N_2 - 1$ , we have  $|(\Lambda A)[j] - (\Lambda \hat{A})[j]| \leq \sum_{0 \leq k + \ell \leq d} |\lambda_{k, \ell}| 2^{-\text{prec}} \leq 1/(2(N + N_1 N_2))$ , from which follows  $\|\Lambda A\|_1 \leq \|\Lambda \hat{A}\|_1 + 2^{-1}$ .

Let  $P$  be as in the statement of the Theorem, and

$$Q(x, t) = \sum_{\substack{0 \leq j_1 \leq N_1 - 1 \\ 0 \leq j_2 \leq N_2 - 1}} q_{j_1, j_2} T_{j_1, [a_1, b_1]}(x) T_{j_2, [a_2, b_2]}(t)$$

be the interpolation polynomial for  $P(ux, v(f(x) + z))$  at the order  $(N_1, N_2)$  pairs of Chebyshev nodes of the first kind. Then, the coordinates of  $\Lambda A_1$  are exactly  $q_{j_1, j_2}$ ,  $0 \leq j_1 \leq N_1 - 1, 0 \leq j_2 \leq N_2 - 1$ .

Proposition 4.2 shows that

$$\begin{aligned} \sup_{\substack{x \in [a_1, b_1] \\ z \in [a_2, b_2] \\ t \in [a_2, b_2]}} |Q(x, z) - P(ux, v(f(x) + t))| &\leq 16 \frac{M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(P) \rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \left(\frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}}\right) \\ &\leq 16 \sum_{0 \leq k + \ell \leq d} |\lambda_{k, \ell}| \frac{u^k M_{\rho_1, a_1, b_1}(x)^k v^\ell M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f(x) + z)^\ell \rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \left(\frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}}\right) \end{aligned}$$

hence

$$\begin{aligned} \sup_{\substack{x \in [a_1, b_1] \\ z \in [a_2, b_2]}} |P(ux, v(f(x) + z))| &\leq \sup_{\substack{x \in [a_1, b_1] \\ z \in [a_2, b_2]}} |Q(x, z)| + \\ 16 \sum_{0 \leq k + \ell \leq d} |\lambda_{k, \ell}| \frac{u^k M_{\rho_1, a_1, b_1}(x)^k v^\ell M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f(x) + z)^\ell \rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \left(\frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}}\right) &\leq \sum_{\substack{0 \leq j_1 \leq N_1 - 1 \\ 0 \leq j_2 \leq N_2 - 1}} |q_{j_1, j_2}| + \\ 16 \sum_{0 \leq k + \ell \leq d} |\lambda_{k, \ell}| \frac{u^k M_{\rho_1, a_1, b_1}(x)^k v^\ell M_{\rho_1, a_1, b_1, \rho_2, a_2, b_2}(f(x) + z)^\ell \rho_1 \rho_2}{(\rho_1 - 1)(\rho_2 - 1)} \left(\frac{1}{\rho_1^{N_1}} + \frac{1}{\rho_2^{N_2}}\right) &= \|\Lambda A\|_1 \leq \|\Lambda \hat{A}\|_1 + 2^{-1} \\ &\leq 2^{-1} + \sqrt{N + N_1 N_2} \|\Lambda \hat{A}\|_2 \text{ thanks to Cauchy-Schwarz inequality} \\ &< 2^{-1} + 1/\sqrt{N + N_1 N_2} \leq 1 \text{ since } N \geq 3, N_1, N_2 \geq 1. \end{aligned}$$

□.

UNIVERSITÉ DE LYON, CNRS, ENS DE LYON, INRIA, UNIVERSITÉ CLAUDE-BERNARD LYON 1,  
LABORATOIRE LIP (UMR 5668), LYON, FRANCE.

*Email address:* `Nicolas.Brisebarre@ens-lyon.fr`

UNIVERSITÉ DE LYON, CNRS, ENS DE LYON, INRIA, UNIVERSITÉ CLAUDE-BERNARD LYON 1,  
LABORATOIRE LIP (UMR 5668), LYON, FRANCE.

*Email address:* `Guillaume.Hanrot@ens-lyon.fr`